AD NUMBER

**ADB010611**

LIMITATION CHANGES

TO:

**Approved for public release; distribution is unlimited.**

FROM:

**Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; 03 OCT 1975. Other requests shall be referred to Air Force Avionics Laboratory, Wright-Patterson AFB, OH 45433.**

AUTHORITY

**AFAL ltr, 12 Sep 1977**

AFAL-TR-75-157

(2)

# HARD PREPROCESSOR STUDY



General Electric Company
Aircraft Equipment Division
Utica, New York 13503

March 1976

TECHNICAL REPORT AFAL-TR-75-157

Final Report for Period 1 February 1975 - 30 June 1975






AIR FORCE AVIONICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
Air Force Systems Command
Wright Patterson Air Force Base, Ohio 45433

NOTICE:

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

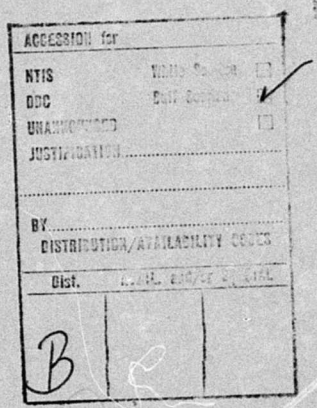
This report has been reviewed and is approved.

Gary L. Mallaley

G. L. Mallaley, 1 Lt, USAF
Project Engineer
Integrated Circuit Design Group
Microelectronics Branch

Stanley E Wagner

Stanley Wagner, Acting Chief
Microelectronics Branch
Electronic Technology Division
Air Force Avionics Laboratory




Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE - 19 APRIL 1976 - 100

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

## REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFAL-TR-75-157
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): HARD PREPROCESSOR STUDY.
5. TYPE OF REPORT & PERIOD COVERED: Final Report. 1 Feb [illegible] - 30 June 1975
6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Dennis E. Berglund, Irving Goodman, William J. Patterson, Julius Reiner
8. CONTRACT OR GRANT NUMBER(s): F33615-75-C-1096
9. PERFORMING ORGANIZATION NAME AND ADDRESS: General Electric Company, Aerospace Electronic Systems Department, Utica, New York 13503
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project 3176 Task 01, Task No. WU12
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Avionics Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio 45433
12. REPORT DATE: March 1976
13. NUMBER OF PAGES: xiii + 124
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 134p.
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution limited to U.S. Government agencies only; test and evaluation results reported: 3 October 1975 Other requests for this document must be referred to the: Air Force Avionics Laboratory (AFAL/DHE) Wright-Patterson Air Force Base, Ohio 45433.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

AF-3176 3176Ø1

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

radiation hardening
microprocessors
preprocessors
computers

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The processing requirements for three preprocessing applications for a microprocessor are determined. The three applications examined are Kalman filtering, radar beamsteering, and the Global Positioning System user equipment computer. The estimated processor requirements include throughput, memory capacity, input/output rates, special instructions and hardware. The device technologies (TTL, ECL, IIL, and MOS) are examined to determine

OVER

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

H02 456

20. (continued)

electrical and radiation performance. The available microprocessors are surveyed and a microprocessor architecture suitable for the possible applications is recommended.

## PREFACE

This report was prepared by the General Electric Company, Aerospace Electronic Systems Department of Utica, New York, under Air Force Contract F33615-74-C-1096, Project No. 3176, Task No. 01. The task is entitled, "Hard Preprocessor Study." The report describes work performed during the period 1 February 1975 through 30 June 1975. The effort was sponsored by the Air Force Avionics Laboratory, Wright Patterson AFB, Ohio. Lt. Gary Mallaley (AFAL/DHE) was contract monitor.

The authors wish to acknowledge the assistance of several individuals who made contributions during the study: Mr. J. T. Doyle, Mr. A. Cooper, Mr. T. Kiesman and Mr. S. Montealegre.

The report was submitted by the authors in October 1975.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

## LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

## LIST OF TABLES

LIST OF TABLES (Continued)

# SECTION I
# INTRODUCTION

Many spaceborne, re-entry, boost ballistic and aircraft digital processors are required to operate in a radiation environment. The use of microprocessors as preprocessors in such applications could result in an increase in performance and flexibility through the use of standard, high density Large Scale Integration (LSI) computing elements. However, the application of presently available microprocessors is restricted to unhardened systems. The solution to this dilemma was the subject of this study which was broken down into three areas: the study of several applications to determine the preprocessor computing requirements; the study of the electrical performance and its degradation in radiation of several device technologies; the study of available microprocessor characteristics, performance, and architecture. The results of these studies are recommendations for the technology and architecture of a microprocessor suitable for use as a preprocessor in a radiation hardened application.

The major sections of this report address the three study areas mentioned previously. Section II describes the computing requirements which were determined for three applications: Kalman filtering, radar beamsteering and the user equipment computer for the Global Positioning System. Section III describes the electrical and radiation performance of the TTL, ECL, IIL, and MOS device technologies. Several examples of the industry-wide hybrid circuit capability are also included. The state-of-the-art in microprocessor technology is reviewed in Section IV. Tradeoff areas in computer architecture are also discussed. The available bit slice devices are compared, and examples of preprocessor architecture are given. Section V contains a summary and the conclusions of the study.

## SECTION II
## APPLICATION STUDIES

### A. NAVIGATION PREPROCESSING

Military navigation systems have different configurations depending on their use and sensor complement. Such factors as manned or unmanned mission, aircraft, space or missile application, tend to influence the types of sensors used, amount of sensor redundancy, precision requirements, form of outputs, etc.

The two basic navigation methods are dead reckoning and position fixing. Dead reckoning involves extrapolation of a known position to some future time, based on the knowledge and/or measurement of position rates, e.g., velocity. Position fixing involves determination of position without reference to any previously derived position. In most systems, both navigation methods are required, since long term dead reckoning would lead to excessive errors without correction fixes. In some cases, involving missile terminal phases, position fixing is either not required, or the dead reckoning and fixing is combined to provide closed loop terminal guidance. The sensor complement generally associated with dead reckoning navigation consists of a true air speed indicator, a doppler or correlation navigator and an inertial measurement unit. Typical position fixing sensors, providing either continuous or periodic fixes, are TACAN, LORAN, OMEGA, GPS (Global Positioning System), star trackers and mapping radars.

In the older systems, the vehicle position, including attitude, if applicable, was usually derived by processing each sensor's data independently, and then either combining the results to reduce the system errors or using them independently to provide redundancy in the case of sensor or processor failure.

New navigation systems, on the other hand, tend to use multisensor processing, usually employing a combinational form of a Kalman filter. This filter, normally implemented in a recursive form, yields an optimum estimate of vehicle state (e.g. position, velocity, attitude, etc.), based on the measurements from all sensors.

The usefulness of the Kalman filter extends beyond the determination of vehicle position coordinates. Since, in the process of execution of the algorithm, the Kalman filter continuously estimates the error variances, these can be used, especially in the multisensor case, for error correction or sensor recalibration.

In the past, Kalman filter algorithms were normally solved as a part of the total navigation problem either in a dedicated Navigation Computer or in the vehicle Central Computer. This has led in many cases to exorbitant software development costs. The present trend, strongly stimulated by the advent of microprocessors, is toward the decentralization of computers so that it is possible that a widespread function such as a Kalman filter will be implemented in a stand-alone processing module. For example, considering the C-5 navigation computer implementation[1*] in Figure 1, the functions within the dashed lines constitute a potential Kalman filter processor. The remaining functions consist of the sensor data conditioning/transformation which could be implemented within the sensor preprocessors, and the horizontal navigation function, which would be either stand alone or a part of a larger navigation function. It should be noted that this form of partitioning allows the use of the Kalman filter processor in either direct or indirect mode, the latter being used in the C-5 aircraft. This approach eliminates the need for any analog-to-digital or digital-to-analog conversion and allows the use of standard data bus at the Kalman filter processor interface.

## 1. Kalman Filter Principles

### a. Mathematics

The use of a Kalman filter in a navigation system provides a means of optimally estimating the vehicle states, e.g. three dimensional position components, velocity components etc. Since the computational algorithm continuously updates the variances of system state errors, additional states such as boresight errors, and gyro drift biases can also be introduced for the purpose of calibration, compensation and alignment.

The Kalman filter algorithm provides the solution to the differential (difference) equations describing the dynamic system of the form

$$X_{n+1} = \Phi_n X_n + U_n \tag{1}$$

The measurements of the system are described by

$$Y_n = H_n X_n + V_n \tag{2}$$

*All references are listed in Appendix A of this document.

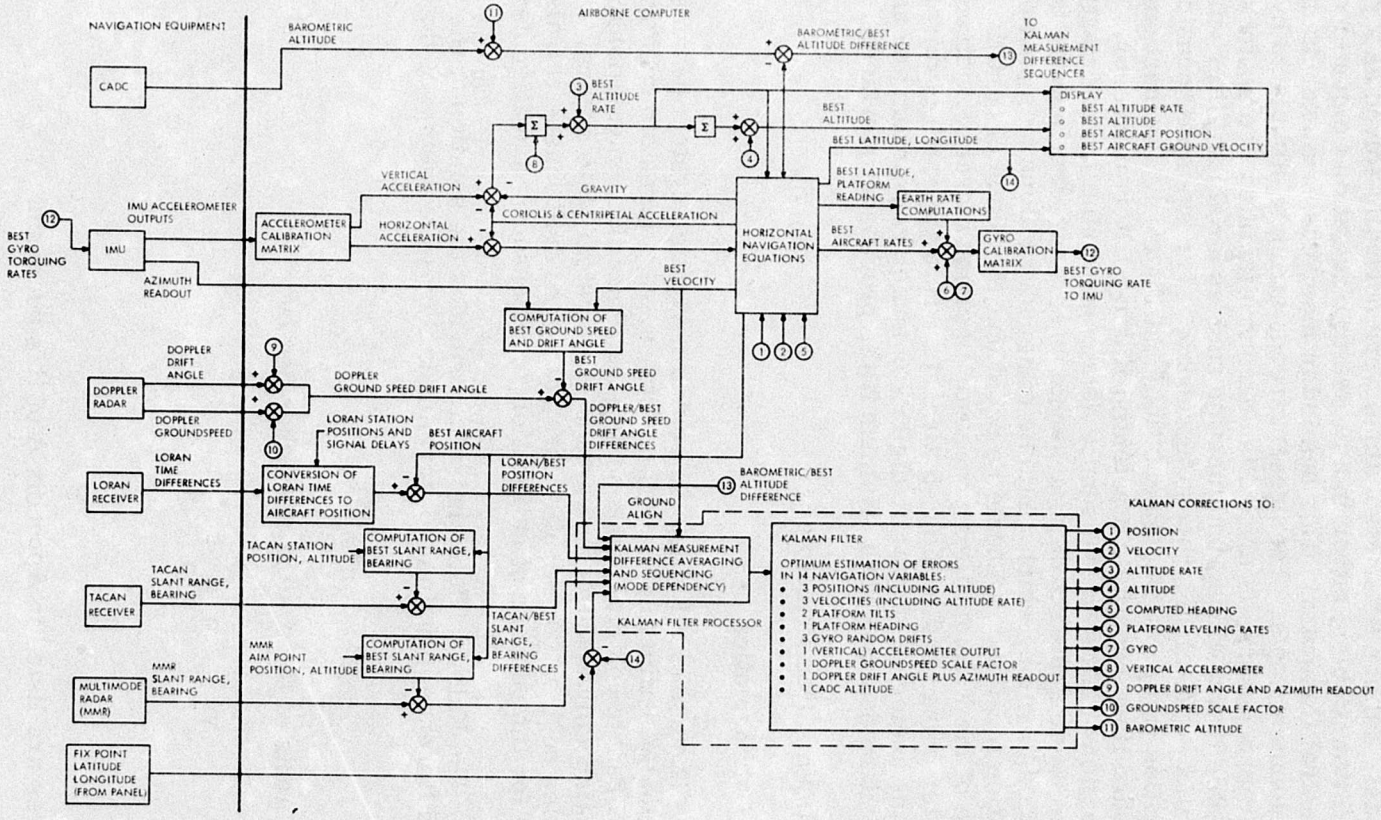



**Figure 1. Functional Block Diagram of C-5 Kalman Navigation/Alignment Mechanism**

where,

$X_n$ = system state vector at time (sample) n

$\Phi_n$ = state transition matrix

$U_n$ = vector of random excitation (referenced to $X_n$ frame)

$Y_n$ = state measurement vector at time n

$H_n$ = measurement transformation matrix

$V_n$ = measurement noise

The above matrix equations are generally solved in the recursive manner, in which the previous estimate of X is combined with a measurement to arrive at a minimum variance estimate.

The optimum estimate $\hat{X}$ of X, given measurements $Y_n$, $Y_{n-1}$, etc. is given by

$$\hat{X}_n = \Phi_{n-1} \hat{X}_{n-1} + K_n \left( Y_n - H_n \Phi_{n-1} \hat{X}_{n-1} \right) \tag{3}$$

where

$K_n$ = gain matrix

$Y_n = -H_n \Phi_{n-1} \hat{X}_{n-1} = r_n$ = residual, or a difference between the new measurements and estimated measurements

The gain matrix, chosen to provide the minimum variance solution is computed from:

$$K_n = \tilde{P}_n H_n^T \left( H_n \tilde{P}_n H_n^T + R_n \right)^{-1} \tag{4}$$

where,

$$\tilde{P}_n = \Phi_{n-1} \hat{P}_{n-1} \Phi_{n-1}^T + Q_{n-1} \tag{5}$$

and

$$\hat{P}_n = \tilde{P}_n - K_n H_n \tilde{P}_n \tag{6}$$

where

$\tilde{P}_n$ = covariance matrix prior to observation (measurement)

$\hat{P}_n$ = estimated covariance matrix after observation

$R_n$ = sensorerror covariance matrix

$Q_n$ = excitation noise covariance matrix

Software implementation of the Kalman filter involves the solution of the matrix equations (3) through (6). A typical execution of these equations is shown in the flow diagram of Figure 2.

Table 1 summarizes the dimensions of Kalman filter matrices and resultant storage requirements.

Typical elements of the state vector, $X_n$, include desired system states and correlated system parameters. For example, the C-5A implementation included the following 16 elements in the state vector:

$$X_n = \begin{bmatrix} \left.\begin{array}{l} \text{3 velocity errors} \\ \text{3 position errors} \\ \text{3 platform altitude errors} \\ \text{3 gyro drift rate errors} \end{array}\right\} \text{Inertial System States} \\ \text{1 accelerometer error} \\ \text{1 doppler ground speed error} \\ \text{1 doppler drift angle error} \\ \text{1 barometric altitude reference error} \end{bmatrix}$$

The procedure used in deriving other Kalman filter matrices is based on expressing the navigation system dynamics, i.e., equations of motion, in terms of these state variables using linear equations. Then, the elements of the transition matrix, $\Phi_n$, are obtained by taking partial derivatives of these equations with respect to the state variables. Similarly, the elements of the measurement matrix, $H_n$, are obtained by taking partial derivatives of equations relating the measurements to system states with respect to the state variables.

The covariance matrix, $P_n$, represents the variance and covariance values of the system states. Frequently an assumption can be made that there is no cross-correlation between the state variables initially. In that case, the initial value of this matrix, $P_0$, will have only diagonal terms, each representing an expected variance of the respective state. Similarly,

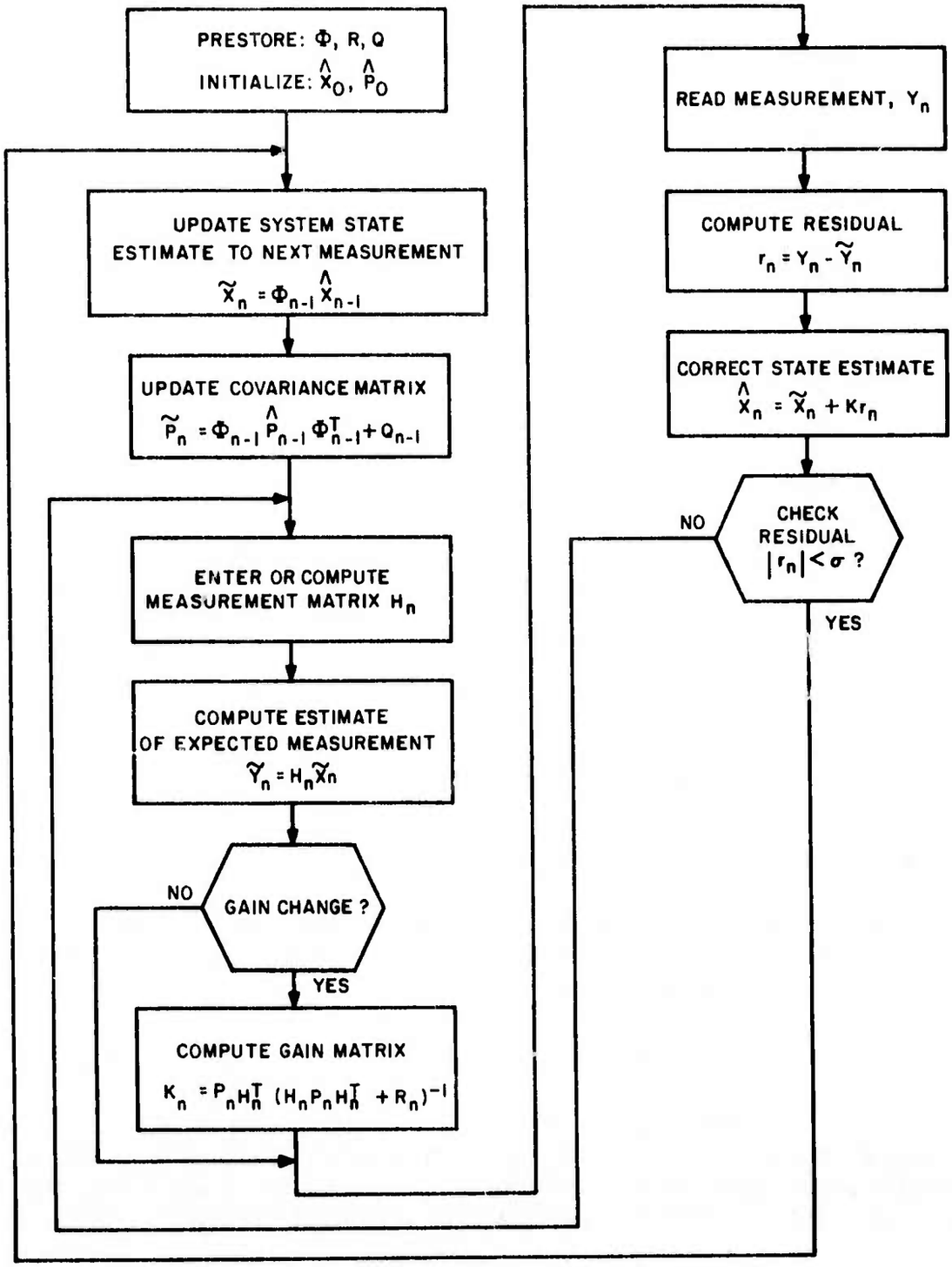



Figure 2. Kalman Filter Program Flow

TABLE 1. KALMAN FILTER MATRICES

| Matrix | Dimension | Storage Requirements (Words) |
|---|---|---|
| $\hat{X}_n$ | n x 1 | n |
| $\tilde{X}_n$ | n x 1 | n |
| $\Phi_n$ | n x n | $n^2$ |
| $\hat{P}_n$ | n x n | $n^2$ |
| $\tilde{P}_n$ | n x n | $n^2$ |
| $Q_n$ | n x n | $n^2$ |
| $H_n$ | r x n | nr |
| $\tilde{Y}_n$ | v 1 | r |
| $Y_n$ | v 1 | r |
| $R_n$ | n x n | $n^2$ |
| $K_n$ | n v | nr |

n = number of states

r = number of measurements

in most applications, it is adequate to assume no cross correlation between different measurements, resulting in the diagonal form of the covariance matrix of the measurement noise, $R_n$.

b. Basic Filter Forms

There are two basic methods of utilizing Kalman filters in navigation systems: either as a direct or an indirect filter. The choice has a considerable impact on the complexity of the resultant algorithms, which in turn affects the amount of computational resources required.

The direct filter implementation, shown in Figure 3, involves direct computation of vehicle parameters (position, velocity and possibly

altitude) by including them in the filter states. This method is commonly used in space navigation, particularly in those systems which do not utilize an inertial platform. In those systems, even though the output states are included in the filter's state vector, the measurements (e.g. star tracker sightings) are usually far apart so that the measurement errors may be considered uncorrelated and no additional state variables need be considered. Thus, in a typical space system, the state vector has six parameters (three position and three velocity components). On the other hand, a direct filter implementation for aircraft navigation yields a very complex solution since additional states must be included due to high frequency correlated errors such as accelerometer bias and gyro drift rates. In addition, filter solution rates are considerably higher since the smoothed vehicle position computations must follow the high frequency components of the vehicle motion.

To circumvent these computational problems, aircraft systems usually employ an indirect filter implementation, shown in Figure 4. In this case the Kalman filter estimates only the error states of the inertial navigation system, rather than the states of the vehicle directly. Since the inertial platform has excellent short term accuracy, filter sampling rates are determined by the navigation augumentation sensors such as LORAN, OMEGA GPS, etc., which may range from seconds to minutes. Thus while the filters for airborne systems are characterized by a large number of states (usually of the order of 10 or more), the solution rates are sufficiently low to allow for low performance computer implementation.

2. Computer Requirements

a. Data Rates and Precision

The Kalman filter processing cycle depends on whether the filter is used directly or indirectly and on the type of reference sensors used. Table 2 lists typical iteration intervals for various sensor types assuming an indirect filter mode in which the filter operates on longer term system state errors rather than following short term vehicle dynamics. Based on that data, one could assume a typical 5 s iteration cycle for establishing processing requirements.

Word precision is directly affected by the system accuracy requirements. Thus 21-bit precision is required to represent a system with 1 s of arc altitude accuracy, neglecting the truncation and round-off errors. Internal processor precision must be higher, considering the covariance matrix computations involving the square of the errors. For example, in the case of the C-5 system, in order to represent the range of squared values of platform tilts (from 1 s of arc to 1°), 24 bits were required not including

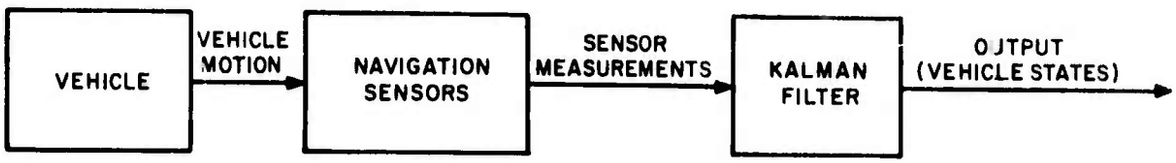



Figure 3. Direct Kalman Filter

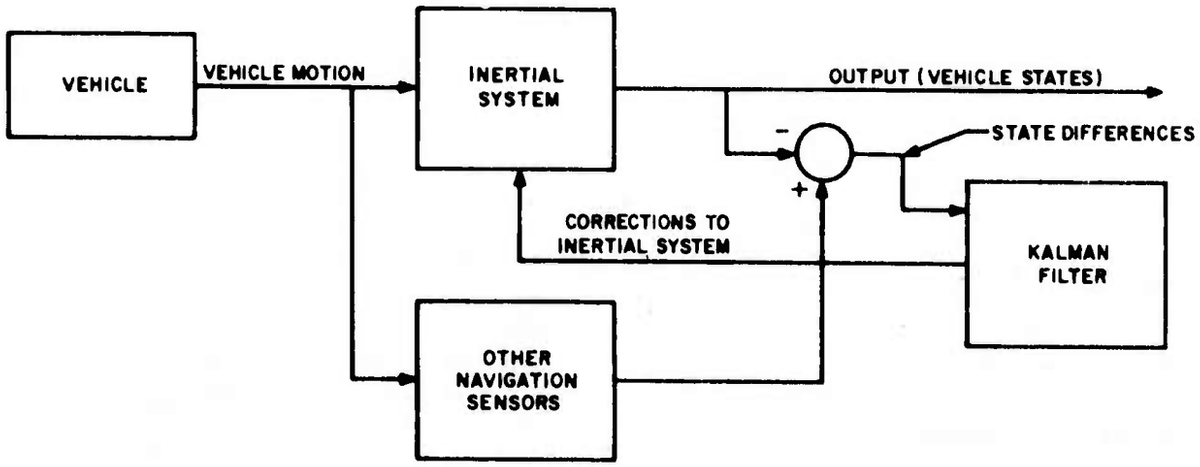



Figure 4. Indirect Kalman Filter

TABLE 2. KALMAN FILTER - ITERATION INTERVALS FOR POSITION CORRECTIONS (FIXES)

| | |
|---|---|
| LORAN/OMEGA | 10 s (after prefiltering) |
| GPS | 6 s |
| Doppler with Strapdown System | 1-1.5 s (worst case) |
| Star Tracker | 6-10 minutes |

truncation/round-off errors. To allow for the latter, in a single precision processor, 28-30 bits are desirable. Double precision and/or floating point capability offers ways of getting around this problem at a cost of hardware complexity. Floating point capability also avoids manual scaling of the data which can be quite tedious because of the multiplicity of matrix operations including inversion.

b. Algorithmic Content Analysis

Examination of Kalman filter equations (3) through (6) and computational flow in Figure 2 reveals the need for an instruction repertoire capable of efficient execution of matrix manipulation problems. Maximum efficiency and speed are required in execution of the following basic instructions:

Load/Store
Add/Subtract
Multiply
Conditional and Unconditional Branches
Indexing (including at least 3 hardwired indexable registers)

Desirable macro instructions or efficient subroutines are as follows:

Matrix Addition/Subtraction
Matrix Multiplication
Matrix Inversion (e.g. Gauss-Jordan algorithm)
Scalar-Vector Product

c. Computational Loading Analysis

J. M. Mendel's[2] article provides a useful method for a quick estimate of Kalman filter computational loading. The results of this analysis are pessimistic in a sense that all matrices and vectors are assumed to be full and that all the computations are performed once per iteration. On the other hand, Mendel's model does not include precomputation of matrix elements and other preprocessing functions which may be required.

The computational loading estimates are expressed in terms of

n = number of state variables

r = number of measurements

s = number of system disturbances

The data is based on programming a hypothetical computer with instruction times as shown in Table 3. It should be noted that the instruction times are expressed in terms of unit times (e.g. memory cycles) except for multiplv, MUL, and divide, DIV, to allow for their wide variations in various computers.

Mendel defines the total computation time, $C(n, r, s, MUL, DIV)$ as follows:

$$C(n, r, s, MUL, DIV) = [M(n, r, s, ) \cdot MUL + 2A(n, r, s) + L(n, r, s, MUL, DIV)]T_u$$

where

$M(n, r, s)$ = total number of multiplications (See Figure 5)

$A(n, r, s)$ = total number of additions (See Figure 6)

$L(n, r, s, MUL, DIV)$ = total logic time (See Figure 7)

$T_u$ = basic unit time (e.g. 1 $\mu$s/unit time)

For example, consider a typical aircraft navigation problem which has 14 states, six measurements and one disturbance. Then from Figures 5, 6 and 7,

TABLE 3. COMPUTER MODEL USED IN ANALYSIS

| Instruction | Execution Time (in unit times) |
|---|---|
| ADD TO A | 2 |
| SUB FROM A | 2 |
| LOAD A OR B | 2 |
| STORE A OR B | 2 |
| MULTIPLY A WITH MEMORY | MUL |
| DIVIDE A BY MEMORY | DIV |
| MARK PLACE AND TRANSFER | 2 |
| RETURN BRANCH | 2 |
| ALL OTHER (E.G. TRANSFERS, INDEX REGISTER INCR.) | 1 |

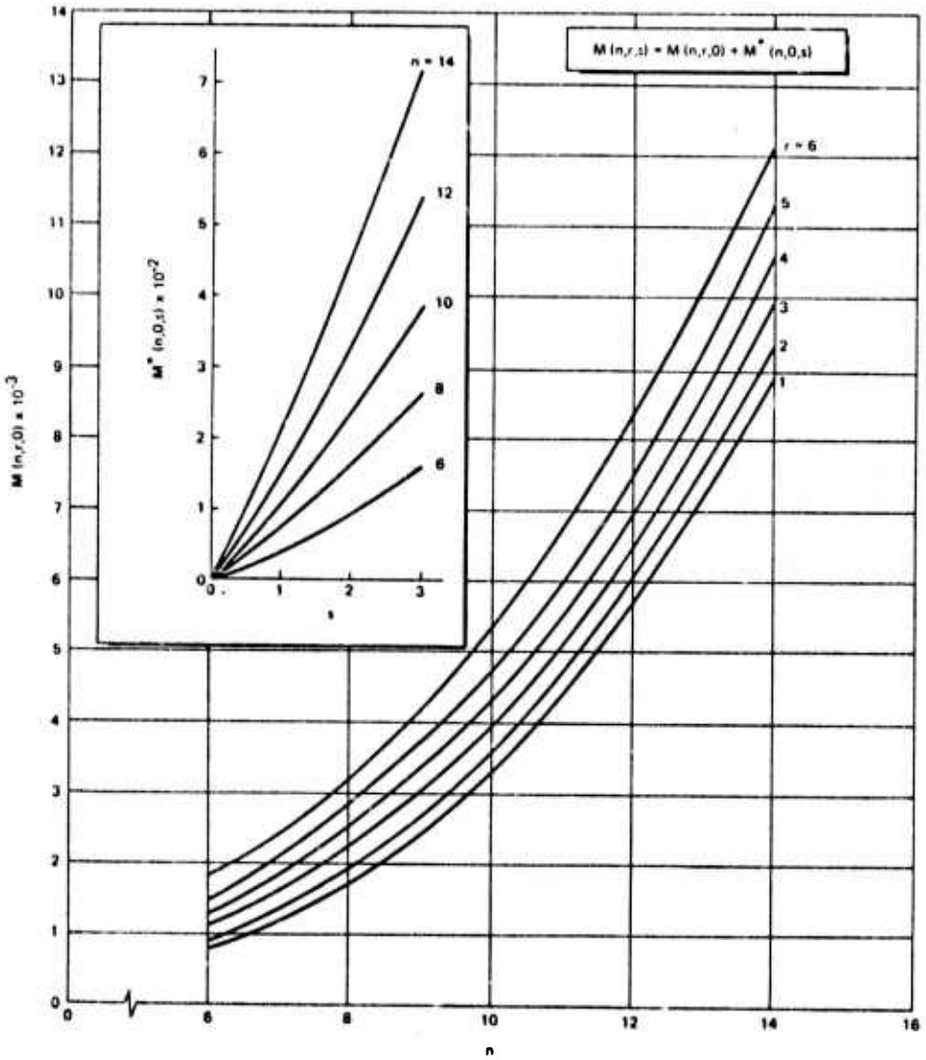



Figure 5. Number of Multiplications M(n, r, s) for a Discrete Kalman Filter

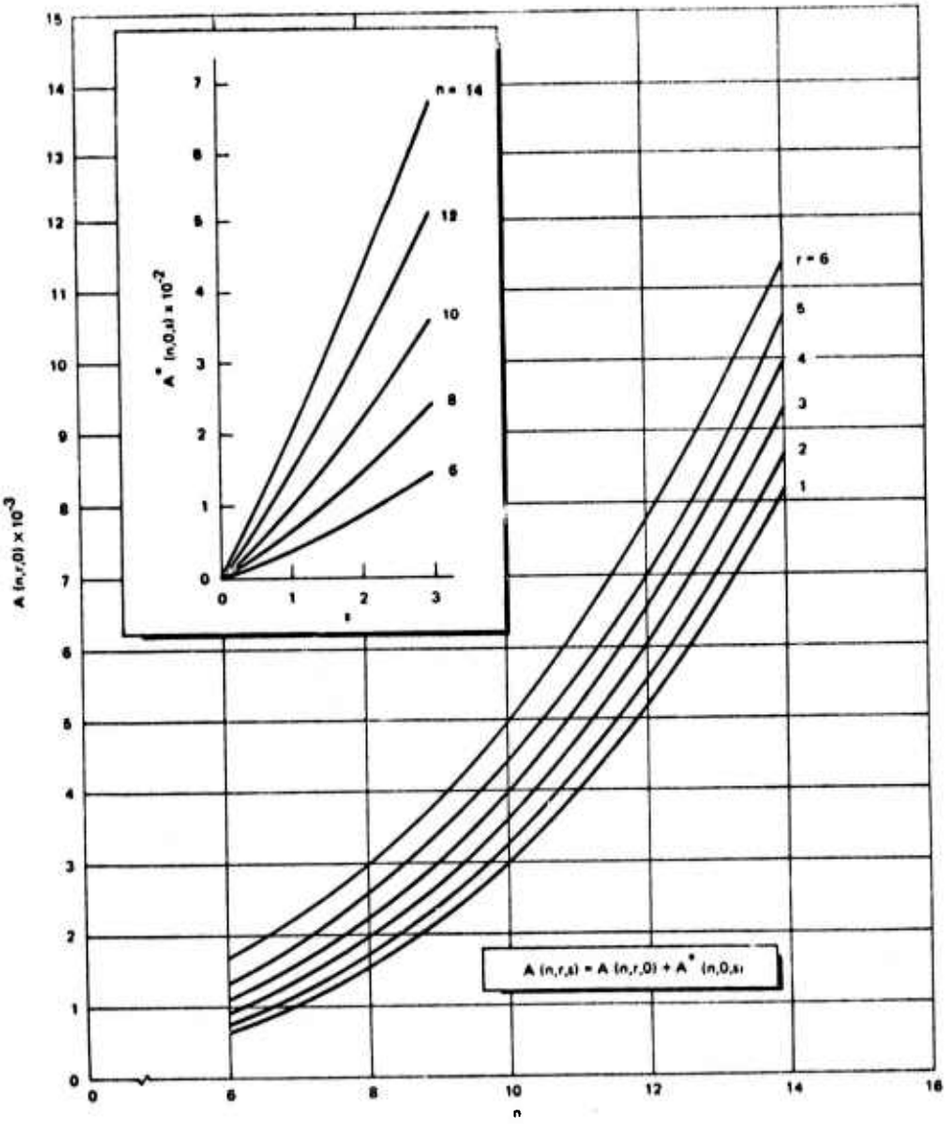



Figure 6. Number of Additions A(n, r, s) for a Discrete Kalman Filter

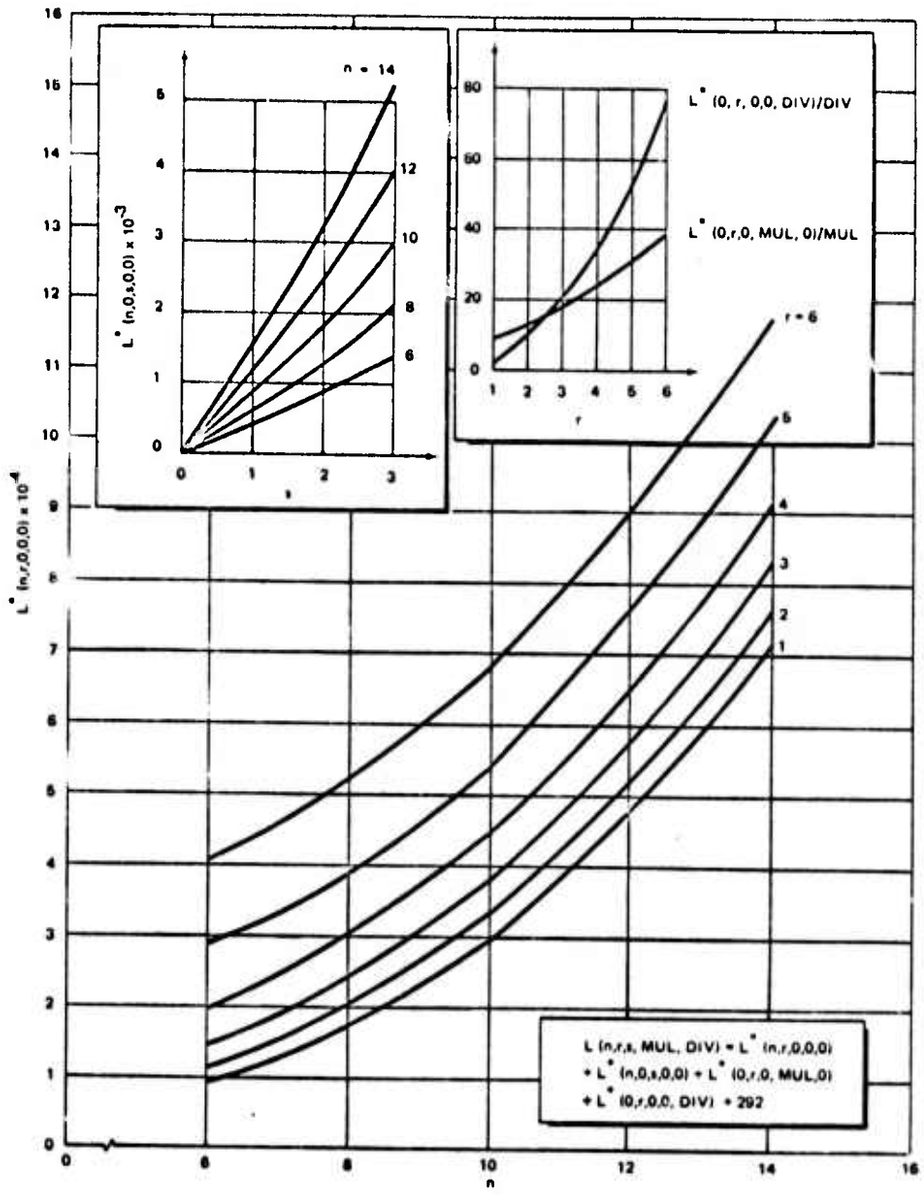



Figure 7. Logic Time in Unit Times L(n, r, s, MUL, DIV) for a Discrete Kalman Filter

$$M(n,r,s) = 12 \times 10^3$$

$$A(n,r,s) = 11.4 \times 10^3$$

$$L(n,r,s,MUL,DIV) = 11.8 \times 10^4 + 1.6 \times 10^3 + 80\ DIV + 35\ MUL$$
$$\cong 11.96 \times 10^4 + 80\ DIV$$

and

$$C(n,r,s,MUL,DIV) = [1.2 \times 10^4\ MUL + 2.28 \times 10^4 + 11.96 \times 10^4 + 80\ DIV]\ T_u$$

Assuming $T_u$ = 1 $\mu$s/unit time (corresponding to 2 $\mu$s add time) one obtains a plot of processing time as a function of the multiply time, shown in Figure 8. This result shows that even present day microprocessors which lack fast multiply capability can be used for a wide range of Kalman filter problems.

In addition, processing loads can be reduced in some systems through the use of such techniques as

(1) Using iteration rates other than covariance and filter gain updating for measurements updating.

(2) Precomputation of filter gain based on a priori data.

d. Storage Requirements

Mendel's article can also be used for estimating the processor storage requirements for solving the Kalman filter problem. Figure 9 plots the total (program and data) storage requirement as a function of n, r and s.

For the preceding example of n = 14, r = 6 and s = 1, one obtains from Figure 9 the following estimate:

$$S(n,r,s) = 1200 + 50 + 489 = 1739 \text{ words}$$

This estimate compares with 220 words of storage allocated to the C-5 Kalman filter problem[3] (in which n = 16, r = 6). In general, in addition to the basic Kalman filter algorithm, some additional preprocessing or post processing functions may be included. Therefore in many systems up to 4K words of memory are usually allotted for this problem.

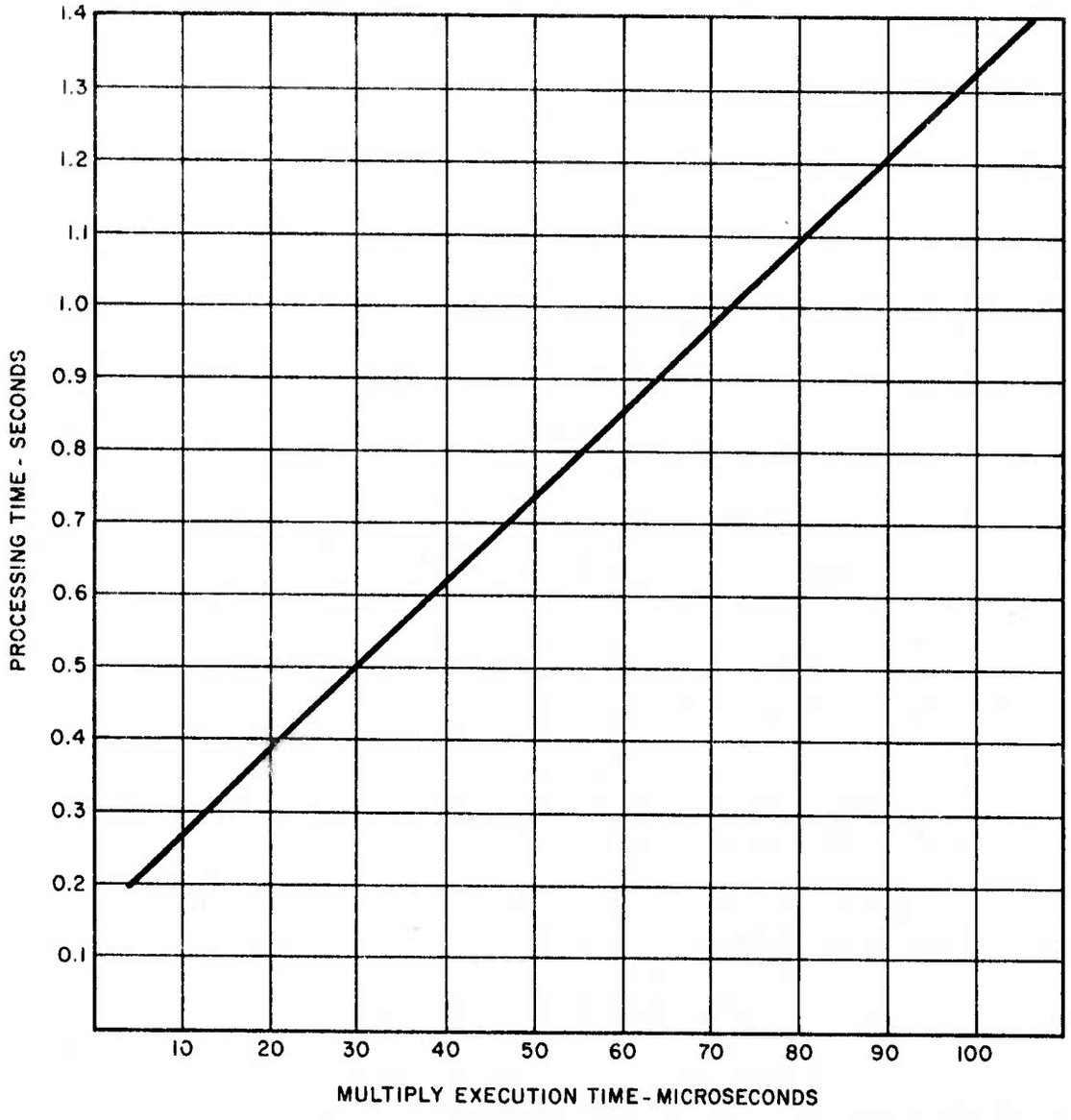



Figure 8. Kalman Filter Processing Time as a Function of Multiply Execution Time

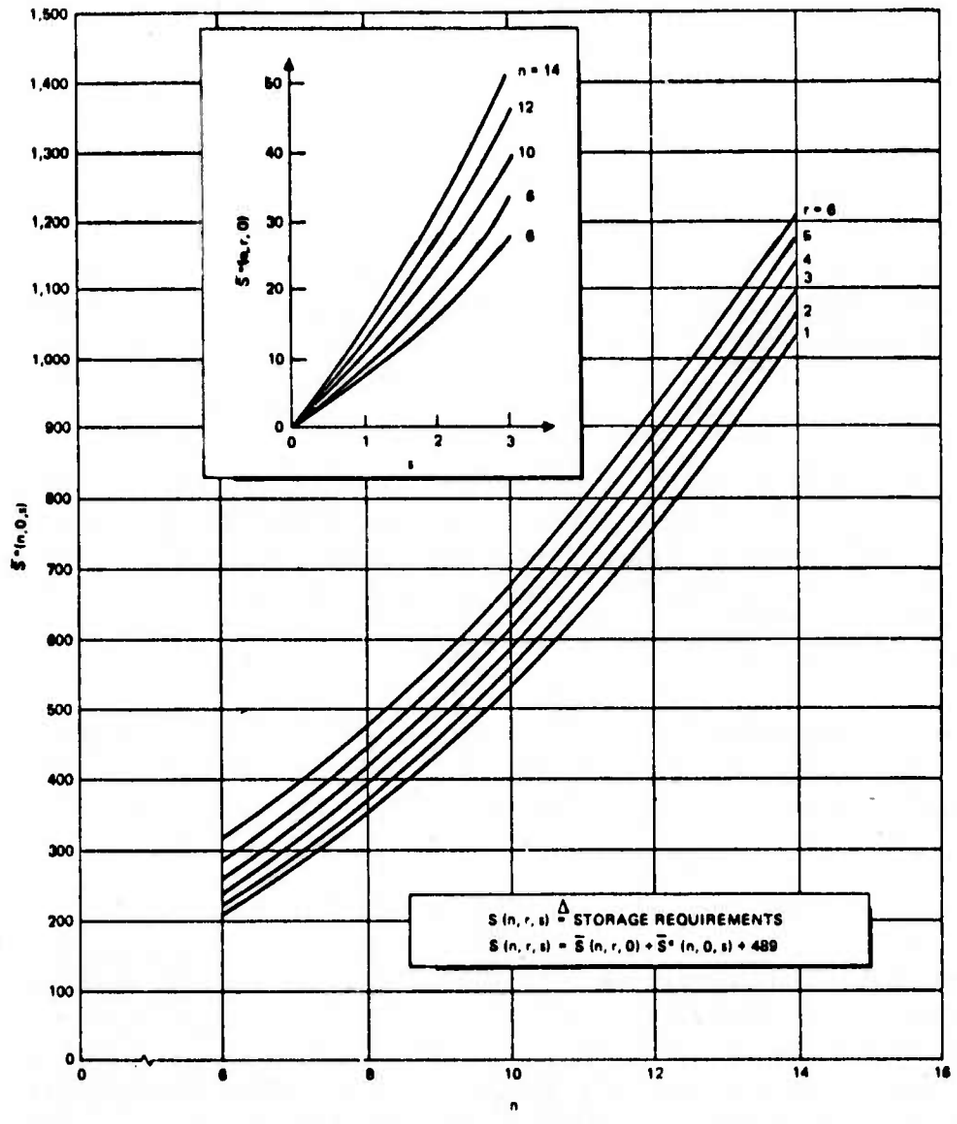



**Figure 9. Storage Requirements, in Cells, for a Discrete Kalman Filter**

e. Summary of Microprocessor Requirements

Based on the preceding analysis, the desirable microprocessor characteristics for the Kalman filter are shown in Table 4. These are based on the 14-state system, assuming an average Kalman filter processing cycle of 1 to 5 s and allows for 30% processor utilization to accommodate some additional peripheral functions. While the required 15 to 100 $\mu$s multiply times are realizable at the upper range of the scale with some available bit-slice microprocessors (e.g. MMI 5701, Intel 3000 series), a fast multiply add-on option may be desirable for more stringent applications. It should be noted that a bit-slice architecture is practically mandatory for such reasons as wide word precision, memory addressing and flexibility.

## B. RADAR APPLICATIONS

The elements in the signal processing chain in a multimode radar are the antenna, transmitter/receiver and signal processor. Associated with each of these elements are corresponding control units performing antenna control, timing and mode control, and processor control. Possible radar applications for microprocessors exist as controllers and signal processors.

As controllers, the microprocessor requirements for the radar application are similar to other equipments. Logical and conditional branching operations are performed more frequently than straightforward arithmetic computations. The controller throughput requirements are low since changes in the external environment, reflected in operator mode changes, are slow and infrequent.

The antenna control function was examined in detail as a typical application of microprocessors in radars. A brief examination of the signal processing requirements was sufficient to deter detailed study in that area.

The processing and control functions which are required in the antenna area are shown in Figure 10. The Scan Control Unit (SCU) examines the command inputs and determines the antenna pointing angles, pattern and polarization. Frequency information from the transmitter exciter is passed on to the Beamsteering Computer. The Stabilization Computer uses the aircraft attitude information to stabilize the pointing angle command. The Beamsteering Computer determines the phase shift weights for each antenna element. The computers need not be separate hardware entities as implied by Figure 10. The processing requirements for the Beamsteering Computer are examined in the following paragraphs. The antenna is assumed to be a phased array.

TABLE 4. KALMAN FILTER PROCESSOR REQUIREMENTS

| | |
|---|---|
| Architecture | Bit-Slice |
| Word Length | 24 bits minimum<br>30 bits (without double precision) |
| Double Precision | Yes |
| Floating Point | Desirable |
| Instruction Execution Times | |
| Add | 2 $\mu$s |
| Multiply (Subroutine) | 15-100$\mu$s |
| Input/Output | 1000 words per second |
| Basic Instruction Set | Add/Sub |
| | Logical, Compare, Branches |
| | Shifts |
| | Subroutine call with return |
| | Fast Multiply Option (5-10 $\mu$s) |
| Desirable Macros | Matrix Add, Multiply<br>Trigonometric (if sensor preprocessing, and other navigation functions included) |
| Addressing Modes | Direct - 4K words<br>Indirect<br>Indexed |
| Number of index (or Indexable) Registers | 3 minimum |

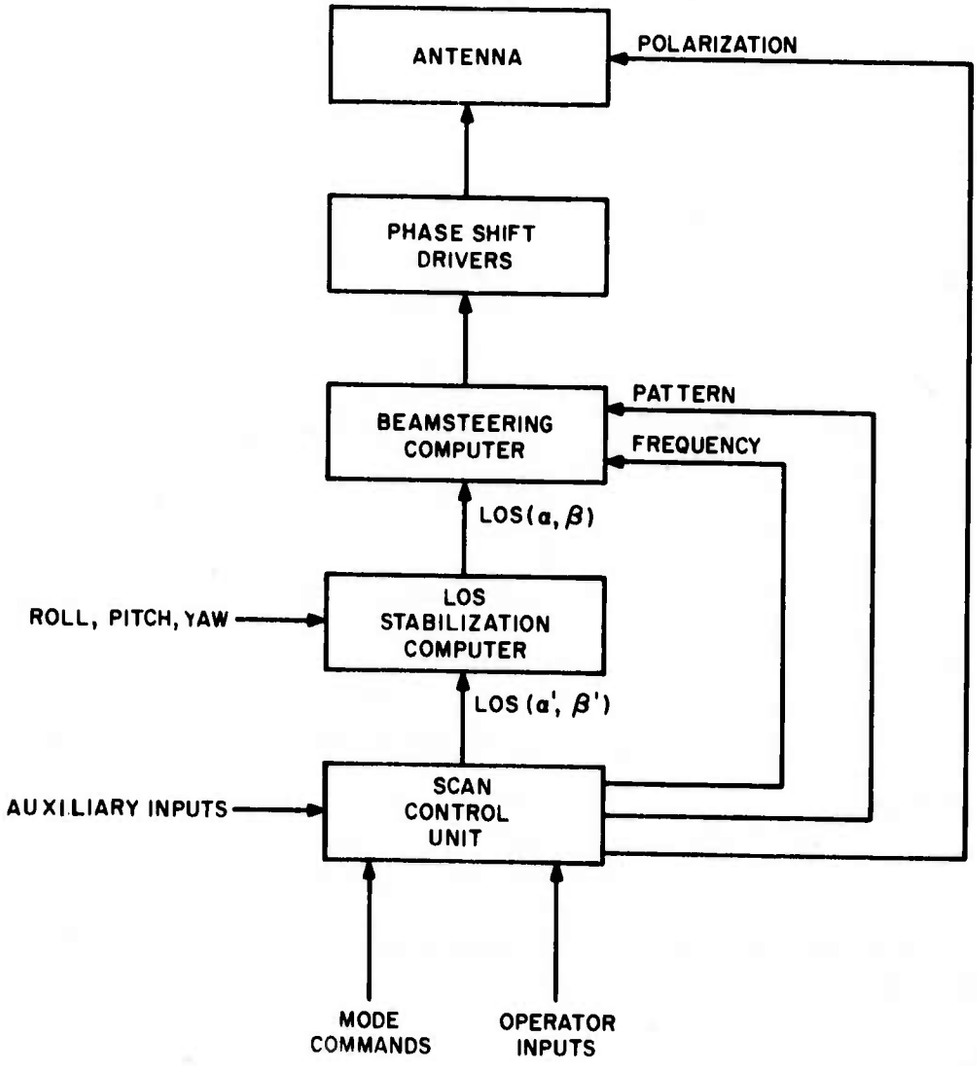



Figure 10. Antenna Control

For a pencil beam and a row and column antenna, the beamsteering computations are straightforward. For each beam position, the x and y axis phase tapers are found from

$$\Delta\phi x = \frac{2\pi\ Dx}{\lambda} \sin\alpha \sin B = a\pi$$

$$\Delta\phi y = \frac{2\pi\ Dy}{\lambda} \sin\alpha \cos B = b\pi$$

where

$\Delta\phi x$ = adjacent element phase shift along x-axis

$\Delta\phi y$ = adjacent element phase shift along y-axis

Dx = element spacing along x-axis

Dy = element spacing along y-axis

$\lambda$ = radar wavelengths

$\alpha, B$ = pointing angles

j, k = element index

The required phase angle for each antenna element is found from

$$\phi(j,k) = j\Delta\phi x + k\Delta\phi y = \pi\ (ja + kb)$$

Typical problem parameters are shown in Table 5. The array consists of a large number of phase shift elements. In the worst case, the antenna position and radar frequency will change on a pulse-to-pulse basis, so that the phase angle of every antenna element would have to be calculated during the pulse repetition interval.

A computer flow diagram for the phase angle computation is shown in Figure 11. The computations are simple: the phase tapers are found from the input variables, and the element phase angles are determined using two indexed loops. The inner loop (k index) is executed for each antenna element and consists of an increment, addition, output and branch operation. Considering the other functions as overhead operation, the computer must complete four operations for every antenna element during every solution interval. The required speed in operations per second is

$$\text{OPS/S} = \frac{\text{OPS/SOLUTION INTERVAL}}{\text{SOLUTION TIME/ELEMENT}} = \frac{4}{\text{PRI/N}} = 12\ \text{MOPS}$$

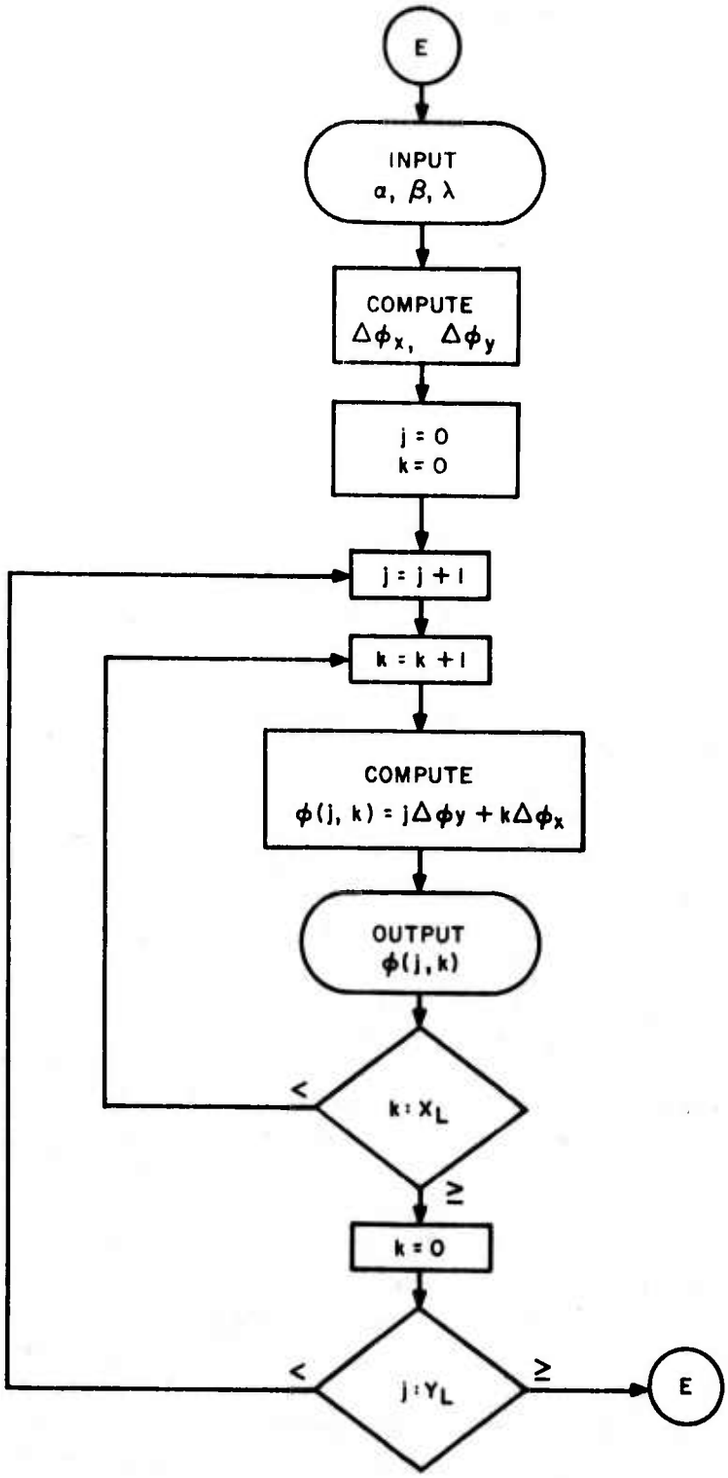



Figure 11. Computer Flow Diagram

TABLE 5. TYPICAL BEAMSTEERING PARAMETERS

| | | |
|---|---|---|
| N | 1500 | Antenna Elements |
| $X_L$ | 50 | x-axis Columns |
| $Y_L$ | 30 | y-axis Rows |
| PRI | 500$\mu$S | Solution Interval |
| m | 5 | Phase Shifter Word Width |
| Dx, Dy | $\lambda/2$ | Interelement Spacing |

Similarly, the output bit rate is

$$\text{OUTPUT RATE} = \frac{\text{NUMBER ELEMENTS (BITS/ELEMENT)}}{\text{INTERPULSE PERIOD}}$$

$$= \frac{Nm}{PRI} = 15 \text{ Mbps}$$

The computer word size may be determined by minimum beam movement, pointing accuracy or truncation considerations. In the latter case, the computer calculations are done with n bit words, while the phase shifter requires m bits. The minimum expressable angle (and the maximum error) in the computer is $2\pi/2^n$. The minimum phase shifter angle is $2\pi/2^m$. If the phase shifts, $\phi(j,k)$, are computed by an addition only algorithm [$\phi(j,k) = \phi(j-1,k) + \pi a$], then the round off error in the $\phi(1,1)$ calculation can propagate to the $\phi(X_L, Y_L)$ element. The maximum error at that element is

$$\Delta\phi = X_L \frac{(2\pi)}{2^n} + Y_L \frac{(2\pi)}{2^n} = (X_L + Y_L) \frac{2\pi}{2^n}$$

This error should be less than the minimum phase shifter angle, $2\pi/2^m$. Therefore,

$$\frac{2\pi}{2^m} > (X_L + Y_L) \frac{(2\pi)}{2^n}$$

or

$$n-m > \log_2 (X_L + Y_L)$$

with the given parameters the minimum word length is

$$n > 5 + \log_2 80 = 5 + 6.32 = 11.32$$

$$n = 12$$

The computer requirements for a single computer solution to the beamsteering problem are summarized in Table 6. Although simple computationally, the solution time per element is quite low (0.33 $\mu$s) and results in very high throughput and I/O rates.

TABLE 6. SINGLE COMPUTER BEAMSTEERING REQUIREMENTS

| | |
|---|---|
| I/O Rate | 15 Mbps |
| Throughput | 12 MOPS |
| Program Memory | 200 Words |
| Temporary Storage | 8 Registers |
| ALU Width | 12 Bits |
| I/O Devices | 1500 |

To reduce the rates to more practical levels, assume that the antenna array is divided into S sections of N/S elements. Assign one computer to each section, so that the system consists of S computers. The increase in hardware reduces the computer requirements as shown in Table 7. For a microprocessor implementation, a computer for each array column (or row) might be a feasible solution.

Radar beamsteering requires very high throughput and I/O rates if the radar requires pulse-to-pulse phase shift computations and the number of antenna elements is large. Lower pulse repetition frequencies, a smaller antenna, or a fixed transmitter frequency would reduce the computer requirements.

TABLE 7. MULTIPLE COMPUTER BEAMSTEERING REQUIREMENTS

| Computer Count | Solution Time/Element $\mu$s | I/O Rate Mbps | Throughput MOPS |
|---|---|---|---|
| 1 | 0.33 | 15 | 12 |
| 3 | 1.00 | 5 | 4 |
| 15 | 5.00 | 1 | 0.75 |
| 30 | 10.00 | 0.5 | 0.40 |
| 45 | 15.00 | 0.3 | 0.27 |

## C. COMMUNICATIONS APPLICATION

The application studied was the user equipment for the Global Positioning System (GPS). The purpose of the GPS is to allow the user to accurately determine his position via satellite communications. The completed system will consist of 24 satellites in different orbital paths above the earth. The user of GPS will be able to receive information from at least four satellites at any given time and location on earth. Each satellite will transmit its precise position at some given time (altitude, longitude and latitude at $t_i$). By knowing their times and positions, the user can calculate his position at a given time (four equations and four unknowns). System characteristics are given in Table 8.

Because of the size, weight and cost objectives, a microprocessor may be the optimum choice for a processor. However, its specification is not trivial. The choice of a processor will greatly affect the size of program memory which is very sensitive to cost. Therefore the data accuracies, speed, and algorithms must be thoroughly defined.

In order to express the altitude of the satellites with an accuracy of several feet, a processor word of at least 24 bits is required. Allowing margin for round off and error propagation, 32-bit arithmetic would be adequate. This precision can be met either with a 32-bit processor or with a 16-bit processor with double precision capability.

TABLE 8. GPS SPECIFICATIONS

| | |
|---|---|
| Total Number of Satellites | 24 |
| Overall Pseudo Noise Sequence Period | 267 days |
| Each Satellite Pseudo Noise Sequence Period (32 unique sequence possibilities) | 7 days |
| Information Frame Time | 6, 12, 18, 24, 30, 36 seconds |
| Number of Bits/Frame (Word) | 300, 600, 900, 1200, 1500, 1800 |
| Transmitting Frequency | 1.5 GHz |
| Maximum Doppler | 3 kHz |
| Perigee Altitude | 20,000 KM |
| Apogee Altitude | 20,000 KM |
| Period | 12 hours |
| Eccentricity | 0-.005 |
| Duration | 5 years |
| Semimajor Axis | 26,500 KM |
| Accuracy (one SIGMA prediction error) | 18 ft. maximum |
| Acquisition Time | 5 minutes |
| Display Update Time | 1 minute |
| Size | 18" x 12" x 6" |
| Weight | 8-25 lbs. |
| Cost | $15,000 to $30,000 |
| Temperature | -40 to +155°F |

The functions to be performed in the user equipment processor are shown in the flow diagram in Figure 12. The user will initialize the processor and estimate his current position and time. Using stored data, the processor will determine the position of all operational satellites, and select a set of four transmitters based on mission performance requirements. The processor will provide data to the receiver for satellite acquisition. This information will include estimated range and doppler and the associated pseudo noise (pn) code which is unique to a given satellite. The satellite data is encoded using a pn code so that a secure communications link can be achieved. The received data is checked and corrected for atmosphere errors, GPS clock errors, and calibrated receiver delays. The position calculation is performed and the results are filtered. The position data is scaled to the appropriate units and formatted for the output display device. The processor then continues to monitor all satellite positions, so that new satellites can be acquired as old ones move out of range. Portions of the software are described in the following paragraphs.

The Satellite Selection Routine requires the calculation of the satellite's positions. The orbits have very small eccentricity, so they can be assumed to be circular. Assuming that the satellite's current period, $T_i$, is known, and that its position at some other time ($\theta_I$ at $t_I$) is also known, then its present position, $\theta$, is

$$\theta_i(t) = \frac{2\pi}{T_i}(t - t_I) + \theta_I \qquad \begin{matrix} i = 1, 2, \ldots 24 \\ 0 \leq \theta < 2\pi \end{matrix}$$

Thus, in its own planar coordinate system, its position $(x_i, y_i)$ is

$$x_i' = R_i \cos \theta_i$$

$$y_i' = R_i \sin \theta_i \qquad \text{where } R_i = \text{orbital radius.}$$

These coordinates can be transformed to the user's inertial coordinate system (or latitude, longitude, altitude) through the appropriate direction cosine transform:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} a_{1i} & a_{2i} \\ a_{3i} & a_{4i} \\ a_{5i} & a_{6i} \end{bmatrix} \begin{bmatrix} x_i' \\ \\ y_i' \end{bmatrix} \qquad i = 1, 2, \ldots 24$$

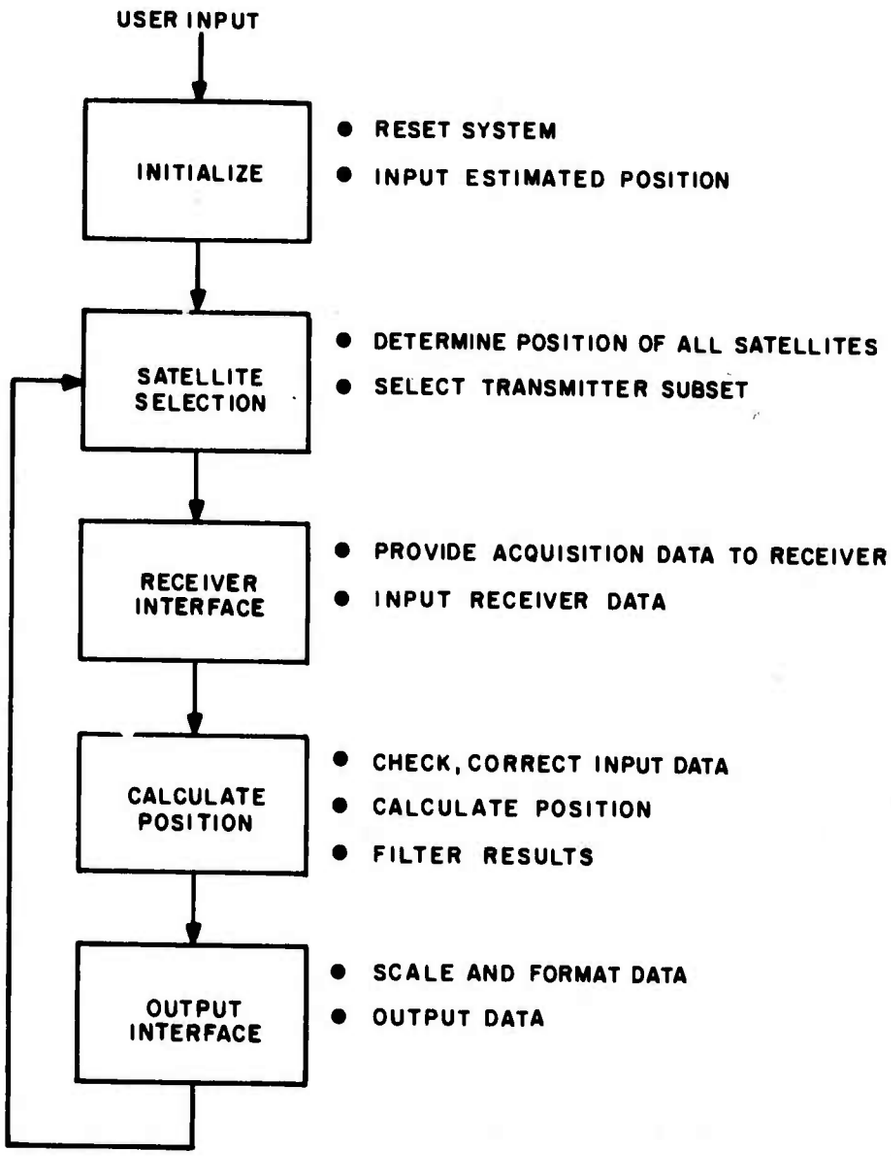



Figure 12. GPS Processor Software

or

$$P_i = A_i P'_i$$

where the $a_{ij}$ are the direction cosine between the plane of the satellite's orbit and the user's coordinate system. This position calculation will require storage of 240 constants (24 x 10) and will result in a position array of 72 entries (24 x 3).

Another portion of the Satellite Selection Routine is the selection of a subset of four of the satellites to provide the position information. The selection criteria could be based on the four closest in range or on the principle called Geometric Dilution of Precision (GDOP). GDOP is a phenomenon which results when the intersecting surfaces which define the user's position are nearly tangent. Although the intersection is a point mathematically, the real world effects of finite precision result in decreased accuracy.

A flow diagram of the Satellite Selection software is shown in Figure 13. Rather than evaluating the selection criterion for each of the 24 satellites, an initial partial ordering is performed which restricts those subsequently examined to the set in the viewable hemisphere. The output of this procedure is a list of four satellites which best satisfy the selection criterion.

The Receiver Interface software examines the selection list and provides the receiver with the estimated range and doppler of the four satellites. The processor also tells the receiver which of 32 possible pn codes each satellite is using.

After the receiver has acquired the four satellites, the user's position can be determined by solving the four pseudo-range equations:

$$\sum_{j=1}^{3} (x_{ij} - u_j)^2 = (r_i - b)^2 \qquad i = 1, 2, 3, 4$$

where

$u_j$ = $j^{\text{th}}$ component of position of the user

$x_{ij}$ = $j^{\text{th}}$ component of position of the $i^{\text{th}}$ satellite

$r_i$ = pseudo range between the $i^{\text{th}}$ satellite and user

$b$ = user's clock bias (units of range)

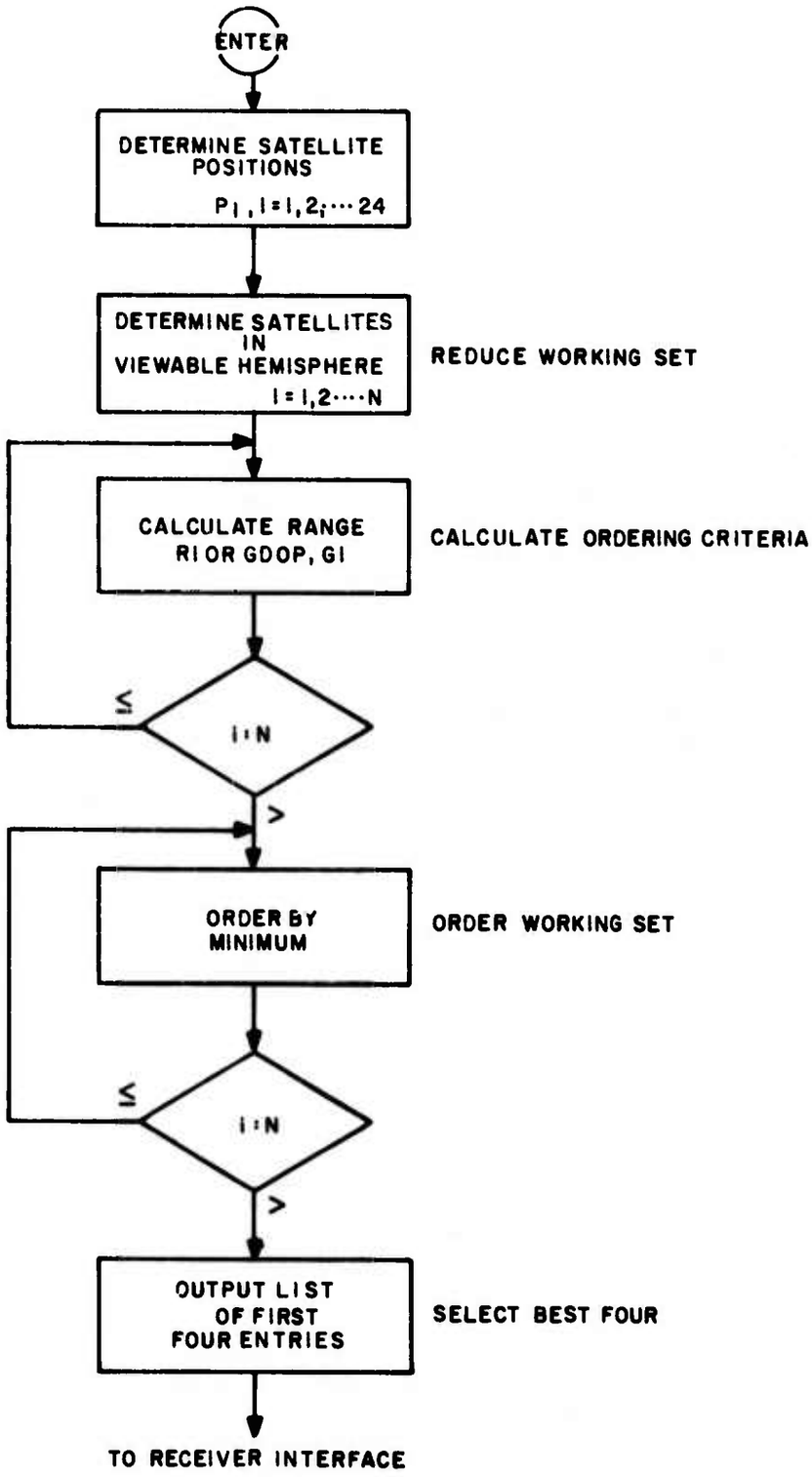



Figure 13. Satellite Selection Software

With the definitions

$$\underline{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \qquad d_i^2 = \sum_{j=1}^{3} x_{ij}^2$$

the last three pseudo range equations can be linearized by subtracting the first, with the result

$$A\underline{u} = b\underline{r} - \frac{1}{2}\underline{v}$$

where

$$A = \begin{bmatrix} x_{21} - x_{11} & x_{22} - x_{12} & x_{23} - x_{13} \\ x_{31} - x_{11} & x_{32} - x_{12} & x_{33} - x_{13} \\ x_{41} - x_{11} & x_{42} - x_{12} & x_{43} - x_{13} \end{bmatrix}$$

$$\underline{r} = \begin{bmatrix} r_2 - r_1 \\ r_3 - r_1 \\ r_4 - r_1 \end{bmatrix}$$

$$\underline{v} = \begin{bmatrix} r_2^2 - r_1^2 + d_1^2 - d_2^2 \\ r_3^2 - r_1^2 + d_1^2 - d_3^2 \\ r_4^2 - r_1^2 + d_1^2 - d_4^2 \end{bmatrix}$$

Thus, the components of $\underline{u}$ can be expressed in terms of the other unknown, b:

$$\underline{u} = bA^{-1}\underline{r} - \frac{1}{2}A^{-1}\underline{v} = b\underline{a} + \underline{c} = b \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

This relation can be substituted in the first psuedo-range equation,

$$\sum_{j=1}^{3} (x_{ij} - u_j)^2 = (r_i - b)^2$$

to obtain a quadratic equation in b,

$$\sum_{j=1}^{3} \left[ x_{ij} - (a_j b + c_j) \right]^2 = (r_1 - b)^2$$

which can be solved using the quadratic formula. The user's position, $\underline{u}$, can then be found by substituting back into the relation

$$\underline{u} = b\underline{a} + \underline{c}$$

This method of solution requires the inversion of a 3 x 3 matrix and a square root operation for each position fix.

The data filter may take many forms. One possible implementation is

$$\underline{f}(n) = T(n, n-1)\,\underline{f}(n-1) + G(n)\underline{u}$$

where

$\underline{f}(n)$ = filtered position output at time n

$T(n, n-1)$ = transition matrix from n-1 to n

$G(n)$ = gain matrix

$\underline{u}$ = current solution

This filter mechanization linearly combines the previous filter output with the new solution point. The matrices may be variable quantities which depend on the system's estimate of the quality of the data.

The computer requirements for the GPS application are summarized in Table 9. The memory storage estimates are shown in detail in Table 10. The throughput and I/O estimates are moderate due to the relatively long time (~6 s) between position inputs. In addition to standard trigonometric and square root subroutines, a matrix inversion procedure is required. Double precision arithmetic is required if a 16 bit processor is used.

The computer requirements have been shown for two types of processors. The first machine has an eight bit instruction and a 32-bit data word. An example of this type of machine is a modified CRD-8. The CRD-8 is a General Electric design of an 8-bit microcomputer that has been constructed using SSI and MSI TTL. For this application, the arithmetic-register section of the Central Processing Unit (CPU) would be extended from eight to 32 bits. When an instruction is read from memory, only the upper eight bits of the 32 bit data path are used to decode the operation. All arithmetic operations use the full 32 bits. Memory space can be minimized by dividing the memory into separate instruction and data sections. However, the user is still limited to an 8 bit instruction.

The second processor is a 16-bit machine with double precision arithmetic capability. The 16 bit instruction word has added capability over an eight bit instruction at the cost of additional memory. The throughput requirement is higher for this machine since more operations are required to arrive at a 32 bit answer. The DEC LSI-11 is a typical microprocessor based machine which could be applied to this application. The LSI-11 can emulate the PDP11-40 minicomputer and it has a floating point option. Extensive software support is also available.

TABLE 9. GPS COMPUTER REQUIREMENTS

| | | |
|---|---|---|
| Memory Size | 10,240 (8 bit) | 6144 (16 bit) |
| Throughput | 100 KOPS | 300 KOPS |
| I/O Rate | 100 Kbps | 100 Kbps |
| Word Size | | |
| Instruction | 8 | 16 |
| Data | 32 | 16 |
| Double Precision | No | Yes |
| Subroutines | Trig<br>Square Root<br>Matrix Inverse | |
| Instruction Set | General Reg Architecture<br>Arithmetic/Logic<br>Mult/Div<br>Subroutines<br>Branch<br>Stack | |
| Address Mod | Direct, Indirect, Index<br>Stack, Register Pointer | |
| Options | Floating Point | |

TABLE 10. MEMORY STORAGE REQUIREMENTS

| | 8-Bit Instruction | 16-Bit Instruction |
|---|---|---|
| Program | | |
| User Input | 128 | 96 |
| Select Four | 256 | 160 |
| RCVR Control | 512 | 425 |
| User Position | 512 | 425 |
| Filter | 1024 | 800 |
| Output | 256 | 256 |
| Control | 512 | 425 |
| Subroutines | 512 | 425 |
| Conversions | 256 | 150 |
| Spec. User Calc. | 512 | 400 |
| Growth, Misc. | 1664 | 534 |
| Constants | 512 (32 bits) | 1024 |
| Scratch pad | 512 (32 bits) | 1024 |
| | 6144 ( 8 bits) | 6144 (16 bits) |
| | 1024 (32 bits) | |
| | 10,240 ( 8 bit bytes) | |

## SECTION III

## TECHNOLOGY STUDIES

### A. PERFORMANCE CHARACTERISTICS

The electrical performance characteristics of the microprocessor candidate technologies are summarized in Table 11, and are discussed in the following paragraphs. Commercial microprocessors are available in all technologies except dielectrically isolated Transistor-Transistor Logic (TTL).

#### 1. Emitter Coupled Logic - ECL

The on-chip propagation delay for ECL 10000 is nominally 2.5 ns which can rise to 3.7 ns over temperature. The off-chip delay of 3.5 ns is with a 50 pf load. The power dissipation includes the terminating resistor dissipation in the off-chip case. The second power supply, -2V, is required for the terminating resistors. Power dissipation is essentially constant over the military temperature range and independent of speed from 1 - 100 MHz. The high level noise immunity is 0.125V and the low level noise immunity is 0.150V at 25°C. Off-chip terminating resistors are required to prevent line ringing. The logic levels for the technology are -0.9 and -1.8V. The negative supply, which represents a departure from normal saturated logic operation, is used to minimize the effects of noise that appears on the voltage supply line, $V_{EE}$ in this case. Normally, the ground bus has the least amount of noise in a system. The logic output levels are closely associated with the $V_{CC}$ terminals, and if minimum noise exists on this line, little effect will be noted on the signal waveshapes. Noise appearing on the $V_{EE}$ supply will be attenuated by about 75% due to the resistor ratios that exist in the circuit. Thus, noise that exists on the $V_{EE}$ line will not have nearly the effect on the signals as equivalent noise on the $V_{CC}$ lines.

A positive 5V power supply can be used if care is taken to minimize noise on the +5V supply line. Translators are available, however, to interface between negative ECL levels and positive saturated-logic levels, and it is normally desirable to operate in this mode. The seemingly arbitrary value of 5.2V was selected to optimize speed/power relationships and to allow three levels of differential-amplifier operation for increased logic efficiency.

The logic swing exhibited by ECL is significantly smaller than that of saturated logic and may seem undesirable to work with. Saturated logic designers are accustomed to typical TTL logic swings of approximately 3 to 3.5 V, while those of ECL are on the order of 850 mV. However, for high-speed

TABLE 11. CANDIDATE TECHNOLOGY PERFORMANCE CHARACTERISTICS

| | | Delay (ns) | Power (mw/gate) | Density ($mils^2$/gate) | Supplies (V) | Noise Immunity (V) | Complexity (masks/diff) | Cost ($/gate) |
|---|---|---|---|---|---|---|---|---|
| ECL 10,000 | On-Chip | 2 | 25 | 120 | -5.2,-2 | 0.125 | 7/5 | 0.75 |
| | Off-Chip | 3.5 | | | | | | |
| STTL | On-Chip | 3 | 15 | 25 | +5 | 0.3 | 7/4 | 0.82 |
| | Off-Chip | 4 | 20 | 75 | | | | |
| IIL | On-Chip | 10 | 0.1 | 10 | +0.85 | 0.5 | 4/2 | ? |
| | Off-Chip | 30 | - | | | | | |
| CMOS (Si Gate) | On-Chip | 15 | 1 | 30 | +10 | 3 | 6/3 | 0.82 |
| | Off-Chip | 45 | 3 | 50 | | | | |
| DI/TTL | | 10 | 10 | | +5 | 0.3 | 7/4 | 5.37 |

operation, smaller signal swings are beneficial from several standpoints. Advantages are that properties of dV/dt and dI/dt are minimized to cut down crosstalk between lines. Further, with smaller signal swings, transmission lines are much easier to handle. The use of transmission lines can be very beneficial at higher speeds.

ECL is a mature technology. The fabrication process requires seven masks and five diffusions. Two layers of interconnect are available: one aluminum layer and one for diffused cross-unders.

The first ECL, bit-slice microprocessor was recently announced by Motorola. The power and silicon area requirements of the technology have undoubtedly contributed to the time span between the ECL and MOS product announcements. The M10800 microprocessor family is microprogrammable, and fast 75 ns instruction cycle times are claimed. Sample quantities are scheduled for early 1976.

2. Schottky TTL

Adding Schottky barrier diodes to the saturating transistors in a basic TTL gate reduces stored charge in the transistors, resulting in higher speed operation. Internal gates in an STTL chip can use fewer transistors than interface gates, resulting in lower power dissipation and less silicon area. The technology is mature, although product lines with different speed power products are available from various manufacturers. The process requires seven masks and four diffusions, and two-layer interconnect is available.

3. Integrated Injection Logic - IIL

IIL is a recent bipolar technology development which offers high packing density and a wide range speed-power tradeoff. An example of the packing density capability of $I^2L$ is shown in Figure 14. The p-injector used to power the flop is indicated along the drawing's horizontal axis. Note that an entire $I^2L$ static flip-flop, which would require six 4-by-4 mil bonding pads interfaced to the outside world, would virtually fit under one of those bonding pads. It is this compact geometry that accounts for the high component density of the $I^2L$ LSI processor designs currently in production. Practical $I^2L$ gates can handle a range of six magnitudes or more of injector current - from picoamperes to microamperes - at speeds ranging from hundreds of microseconds to tens of nanoseconds. They can be powered up for maximum speed, then powered down by a magnitude of 100 to 10,000 without losing functions or data (if they are memories). They do not display increased power dissipation with frequency nor produce the switching noise transients common in standard push-pull logic built with CMOS or TTL. They require neither gold doping nor

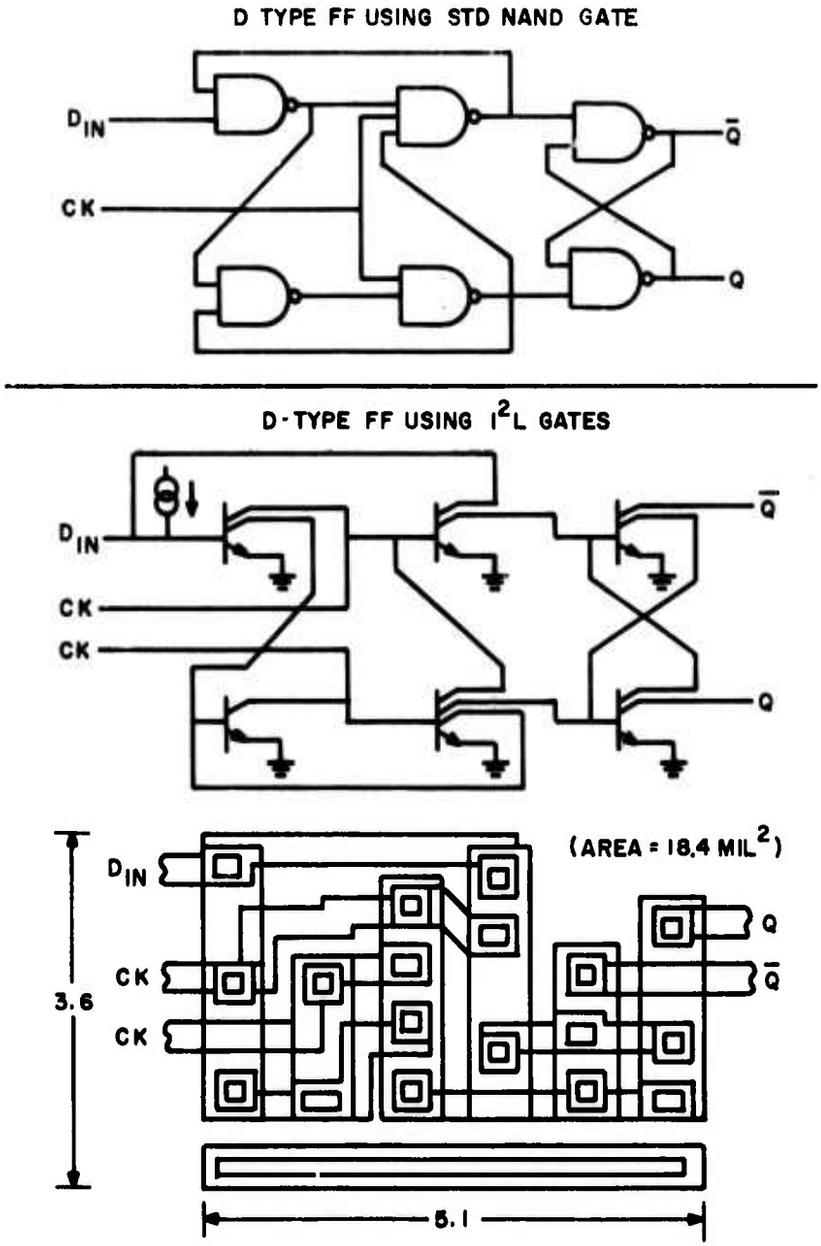



Figure 14. IIL Design Example

Schottky clamping, as does conventional logic in which transistors are easily saturated.

$I^2L$ logic is fully static, requiring no multiphase clocks. Temperature stability is superior, with circuits capable of military temperature range operation from -55°C to 125°C.

Full TTL compatibility can be achieved with the circuits shown in Figure 15. A nominal input threshold of 1.5V can be achieved with two 10K Ω resistors functioning as a voltage divider.

These high-impedance, high-threshold characteristics were chosen to reduce input loading and to increase the input noise margin over a standard TTL input, yet they retain full compatibility with all 5-V logic families. The $I^2L$ inputs also utilize an input-clamping diode to limit negative excursions, or ringing, on the receiving end of a transmission line.

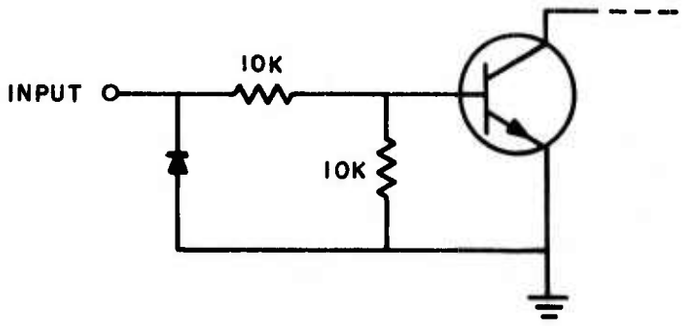
Standard design practice for $I^2L$ is again patterned after TTL in that $I^2L$ outputs are generally over-designed by 100%. While a typical $I^2L$ output will sink 40 mA without pulling out of saturation, the outputs of $I^2L$ logic circuits are guaranteed to sink 20 mA (10 Schottky TTL loads) at 400 mV maximum under worst-case conditions. The high logic output level, output rise times, and input noise immunity are determined by the discrete pull-up resistor.

$I^2L$ is relatively easy to process, requiring only four masks and two diffusions. Although a commercial logic family is not available, the high packing density makes the technology a natural for LSI. The Texas Instruments microprocessor, SBP0400, contains more than 1450 gates.

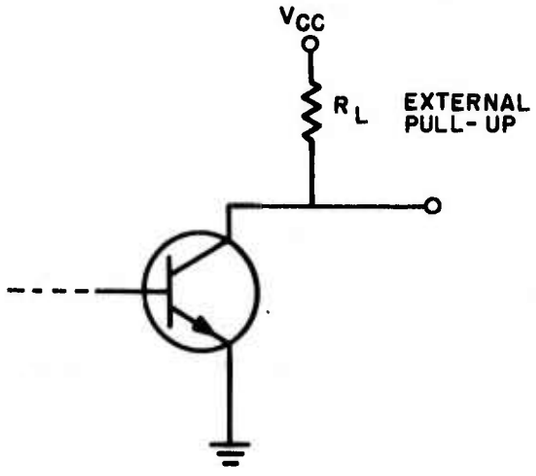
4. Complementary MOS - CMOS

The on-chip propagation delay in CMOS circuits is dependent on the gate capacitive loading. Power dissipation is dynamic in nature, depending directly on capacitive load and switching frequency. Static power dissipation is very low, which makes CMOS very attractive for low speed, battery powered applications. Noise immunity is excellent with 30% of the supply voltage guaranteed. The CMOS, silicon gate process is a mature technology; processing requires six masks and three diffusions.

Sapphire can also be used as a substrate material for CMOS circuits, instead of the conventional bulk silicon. The great advantage of CMOS on SOS is that it can be built into configurations that are two to three times denser than bulk silicon circuits, with two to three times more speed. Moreover, power dissipation at higher speeds is greatly reduced, yielding a




a) INPUT EQUIVALENT CIRCUIT




b) OUTPUT CIRCUIT

Figure 15. TTL Compatibility

speed-power product unmatched by any other technology except integrated injection logic. The implication is clear - a high-performing family of circuits suitable both for LSI logic applications in calculators, processors, and controllers, and for mainstream SSI and MSI standard-logic applications.

The entries in Table 11 show that the highest speed is obtained with the bipolar technologies, at the expense of power and silicon area. Bipolar LSI has been limited to approximately one fourth the size of MOS LSI. Whereas a 4K bit RAM has been demonstrated with MOS, bipolar Schottky has a 1K and ECL a 256 bit memory device. The introduction of low-power Schottky allowed a density increase to 500-1000 gates from the 100-200 gate limit with standard Schottky. The low-power Schottky process combines the features of high speed, good density and moderate power dissipation.

The costs given in Table 11 are based on a two input gate, military temperature range, ceramic package in 100 or more quantities. Although no costs are available for $I^2L$, there is no reason to suspect that it would not be competitive with the other technologies. The most glaring cost entry is for dielectrically isolated TTL, approximately six times the gate cost of the other technologies.

## B. RADIATION PERFORMANCE

The effects of radiation on both bipolar and field effect devices were examined and the performance levels of the various technologies are summarized in Table 12. Supporting data for the levels given in Table 12 were obtained from references with the exception of the ECL 10,000 data which is based on unhardened MECL III data[4], and the NMOS data which is based on PMOS circuits and NMOS devices in CMOS circuits. The IIL data is relatively recent and more should be available in the future. Although no direct data for ECL 10,000 was found, data for a similar technology, Emitter Follower Logic (EFL) was available[5], and is included in Table 12. The following paragraphs discuss the damage mechanisms and the radiation performance of the various technologies in detail.

### 1. Bipolar Technologies

#### a. Damage Mechanisms

Irradiation of bipolar transistors causes:

(1) space charge accumulation in the passivation layer and interface state generation,

TABLE 12. CANDIDATE TECHNOLOGY RADIATION CHARACTERISTICS

| Technology | Neutron Fluence ($N/cm^2$) | Peak Dose Rate: Work Through (Rads(Si)/s) | Peak Dose Rate: Survive (Rads(Si)/s) | Total Dose (Rads(Si)) |
|---|---|---|---|---|
| ECL | $2 \times 10^{15}$ | $1\text{-}3 \times 10^{8}$ | $>1.5 \times 10^{11}$ | $>6.6 \times 10^{6}$ |
| EFL | $> 10^{14}$ | $1\text{-}4 \times 10^{8}$ | $>9 \times 10^{10}$ | $> 4 \times 10^{7}$ |
| IIL | $5\text{-}30 \times 10^{12}$ | $1\text{-}5 \times 10^{8}$ | ? | $10^{4}$ - $10^{5}$ |
| TTL, Hard SSI | $5 \times 10^{14}$ | $> 10^{9}$ | $>10^{12}$ | $> 10^{6}$ |
| MSI/LSI | $> 3 \times 10^{14}$ | $1\text{-}30 \times 10^{8}$ (narrow)<br>$0.02\text{-}20 \times 10^{8}$ (wide) | $6\text{-}30 \times 10^{12}$ | $> 10^{6}$ |
| CMOS, Hard | $>10^{15}$ | $10^{8}$ | $>10^{10}$ | $10^{7}$ |
| CMOS/SOS | $>10^{15}$ | $>10^{10}$ | $>10^{12}$ | $< 3 \times 10^{5}$ |
| PMOS | $>10^{15}$ | $10^{8}$ | $>10^{11}$ | $> 3 \times 10^{6}$ |
| PMOS/SOS | $>10^{15}$ | $>10^{10}$ | $>10^{12}$ | $< 3 \times 10^{5}$ |
| NMOS | $>10^{15}$ | $>10^{8}$ | $>10^{10}$ | ENH. $10^{5}$<br>DEP. $>3 \times 10^{6}$ |
| NMOS/SOS | $>10^{15}$ | $>10^{10}$ | $>10^{12}$ | $3 \times 10^{5}$ |

(2) majority carrier removal at the radiation induced defects,

(3) ionization of the semiconductor material in the vicinity of the circuit p-n junctions.

Circuit performance degrades due to:

(1) transistor gain degradation,

(2) bulk silicon resistivity increase,

(3) transient photocurrent generation causing transient or possibly permanent circuit malfunction.

Three different factors contribute to transistor gain degradation in bipolar circuits. The first two occur as a result of neutron bombardment. The first mechanism is a reduction in the minority carrier lifetime in the base region. In order for the transistor to remain functional, the base transit time must be much shorter than the minority carrier lifetime. As the neutron fluence increases, the minority carrier lifetime approaches the base transit time, eventually reducing the gain of the transistor to zero. A second mechanism is the increase in base current resulting from increased carrier recombination in the emitter space charge region. Normally, this mechanism is dominant only at low injection levels. A third mechanism is the increased surface recombination resulting from an increase in interface state density. Both of these latter mechanisms are especially important at high fluences ($>10^{13}$ N/cm$^2$) for IIL.

Gain degradation with neutron fluence is shown in Figure 16 for several transistors with different gain-bandwidth products ($f_T$). Transistors with high $f_T$ usually have smaller geometries and shorter base widths, so that a higher fluence is required to make the minority carrier lifetime approach the base transit time.

Defects formed by neutron bombardment in bulk silicon tend to act as trapping and recombination centers, thereby increasing the bulk resistivity. The effective majority carrier concentrations tend to decrease and approach the intrinsic level at a rate proportional to the neutron fluence. The bulk resistivity is inversely related to the doping level, so that bulk resistivity increases with neutron fluence. An example of this effect is shown in Figure 17, in which the initial donor concentration in the n-type material was $2.5 \times 10^{15}$ cm$^{-3}$, corresponding to a resistivity of 2 ohm-centimeter. The material becomes intrinsic with a resistivity of $2.2 \times 10^5$ ohm-cm at a critical neutron fluence of approximately $5 \times 10^{14}$ neut/cm$^2$. A minimum doping concentration can be determined if the neutron fluence for the specified environment

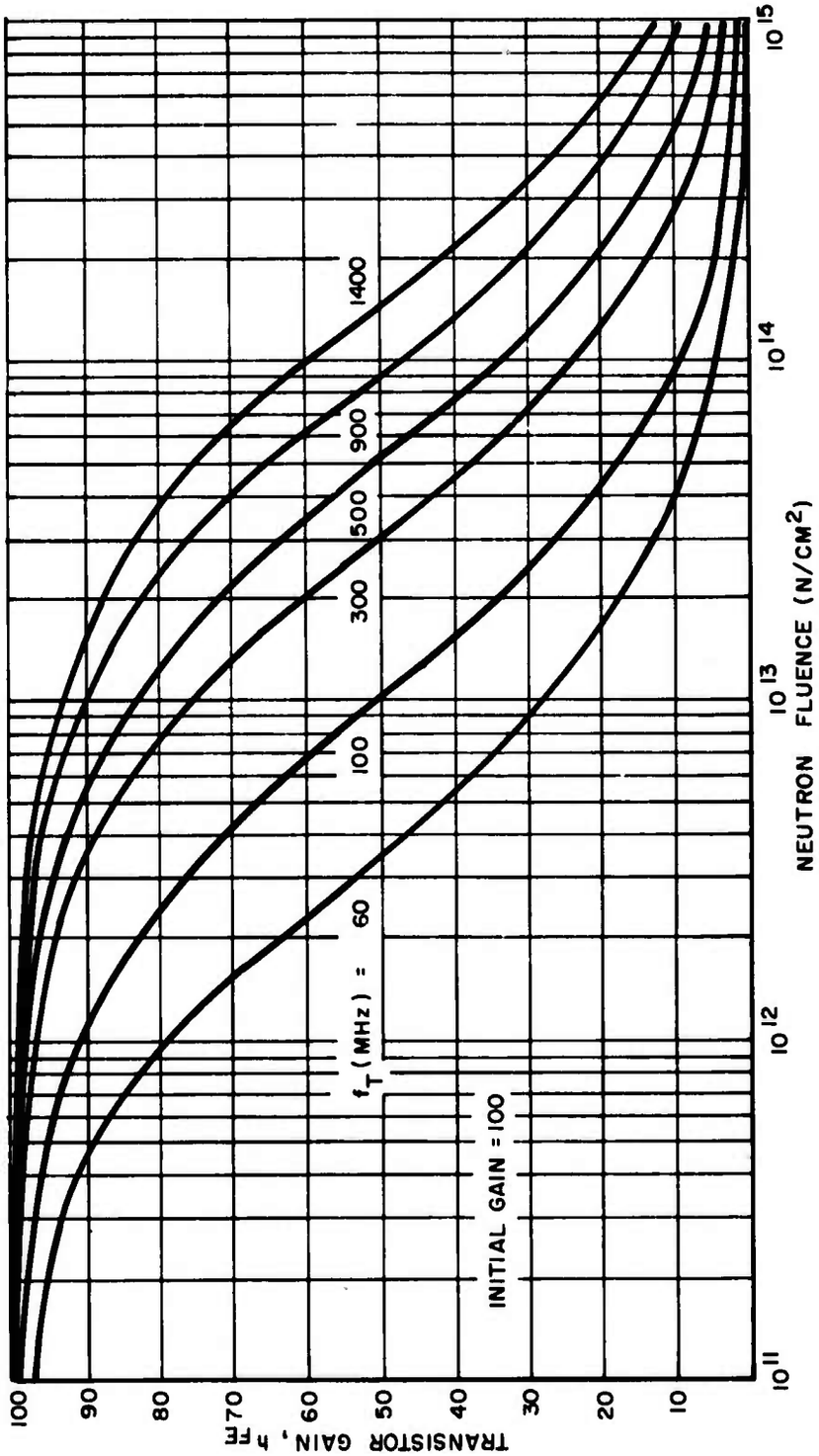



Figure 16. Bipolar Transistor Gain Degradation

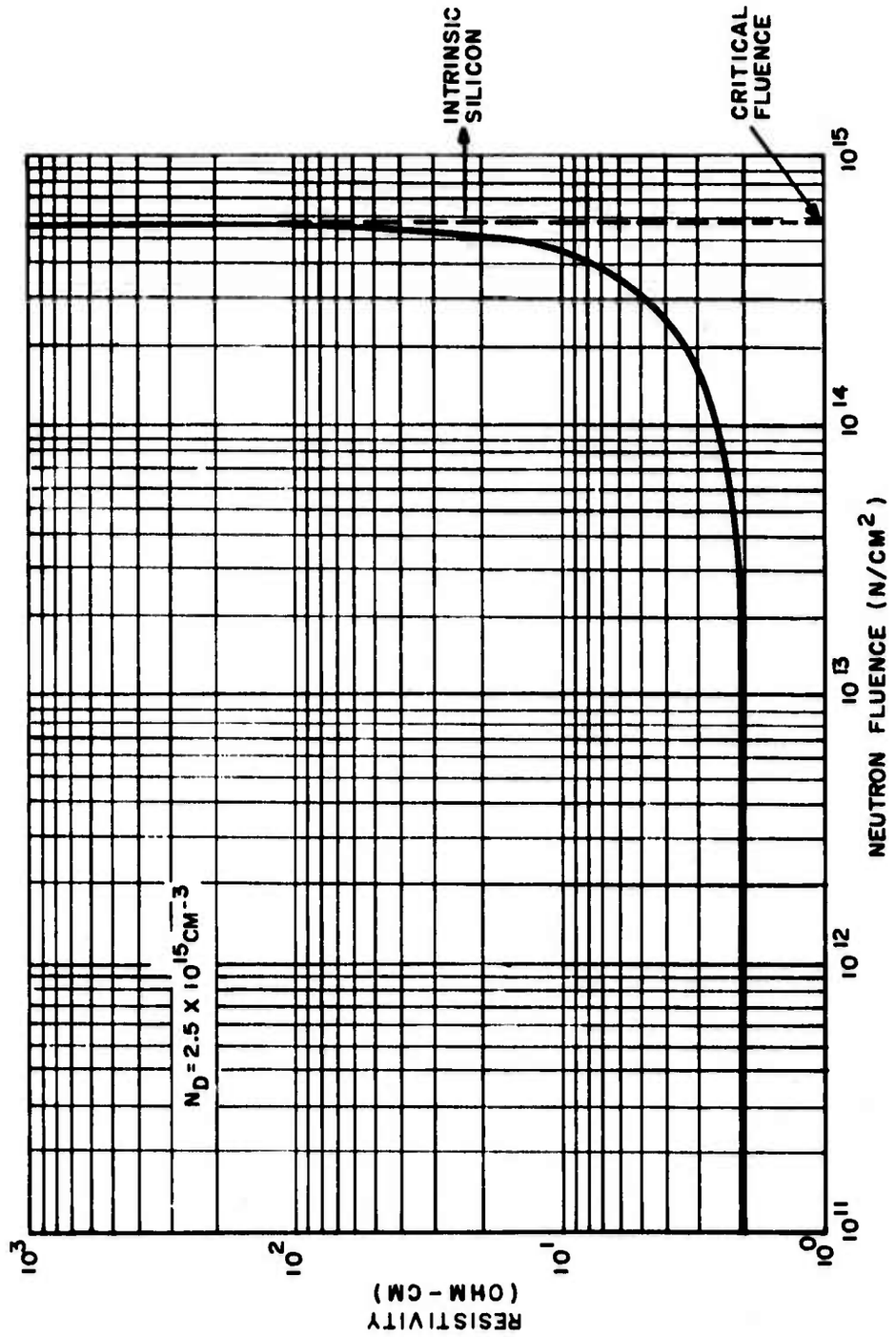



Figure 17. Neutron Fluence Effect on Resistivity

is known. For example, if the specified fluence level is $10^{14}$ neut/cm$^2$, the minimum donor level is just under $10^{15}$ cm$^{-3}$ in vacuum float zone silicon and approximately 5 x $10^{14}$ cm$^{-3}$ in Czochralski silicon.

High levels of ionizing radiation generate electron hole pairs in the vicinity of each p-n junction, resulting in transient currents which can charge and discharge circuit nodes. The transient current is proportional to the p-n junction area and the transient dose rate. Computer models can be used to simulate the effects of transient radiation in bipolar and field effect circuits.

b. Logic Family Performance

(1) Emitter Coupled Logic

There are several potential problems associated with the use of ECL circuits in a radiation environment.

(a) They are not available off-the-shelf in a dielectrically isolated form, even in small scale integrated circuits.

(b) The input base resistor is high (see Figure 18) so that a relatively low photocurrent through the resistor will bias the input transistor on, possibly resulting in an erroneous output.

(c) All transistors operate in their linear region, making the circuit more susceptible to $h_{FE}$ degradation.

(d) The logic levels are separated by only 800-900 mV and the relatively low, 0.125 V, noise immunity is likely to be aggravated under radiation exposure.

The MC 1679L, a four bit MECL III counter, has been tested in various radiation environments[4]. The circuit, shown in Figure 19, can perform the divide by two, divide by five, or divide by ten functions.

A divide by ten configuration was used in FXR and LINAC tests; the lowest failure level was obtained when all bits of the counter were in a logic "1" state. When exposed, the "1's" changed to "0's" at a level of 3 x $10^8$ Rads(Si)/s for all seven samples tested. No failures were observed at 2 x $10^8$ Rads(Si)/s and identical performance was obtained for both static and dynamic operations (920 MHz clock rate). In the wide pulse (4 $\mu$s) LINAC tests, the devices failed by dropping "1" between 1 and 2 x $10^8$ Rads(Si)/s, and when operated at a 2.5 MHz rate, they failed to count between 1.1 - 1.4 $10^8$ Rads(Si)/s.

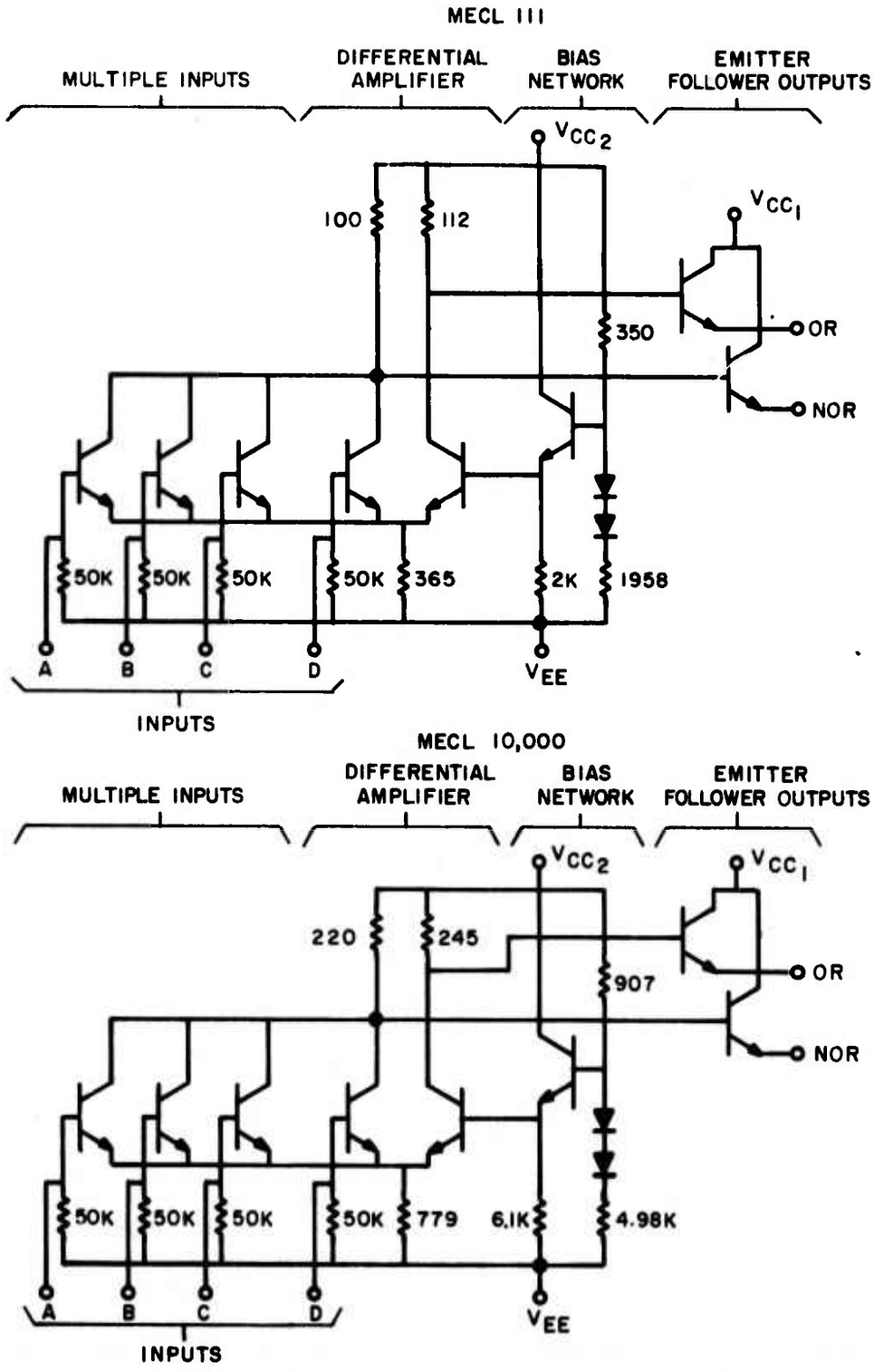



Figure 18. Basic Gate Diagrams for ECL Families

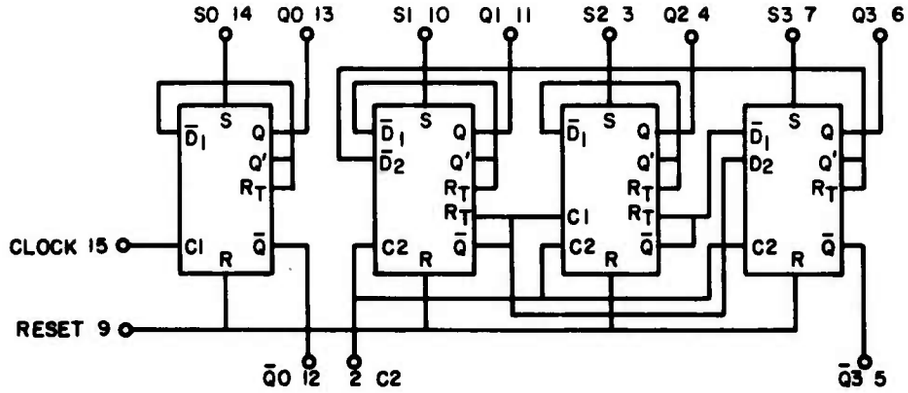



Figure 19. MC 1678 Counter

The circuit also exhibited unexpectedly long recovery times. After exposure to $10^{11}$ Rads(Si)/s, the circuit did not resume counting correctly until 1 $\mu$s after the burst. When exposed to a 4.5 $\mu$s LINAC pulse at $2 \times 10^9$ Rads(Si)/s, the circuit required 14 $\mu$s to recover. No catastrophic damage was observed at the GAMBLE II test facility at a dose rate up to $1.5 \times 10^{11}$ Rads(Si)/s, the highest actually attained.

The circuit operated amazingly well in the neutron environment, to which several were exposed. The observed failure level was $1.9\text{-}2.5 \times 10^{15}$ N/cm$^2$. In failing, the logic level separation decreased from 900 mV to 300 at $1.9 \times 10^{15}$ N/cm$^2$, and the circuit stopped counting, in some cases abruptly.

(2) Emitter Follower Logic

Emitter Follower Logic operates at medium speeds (2-40 MHz), and was initially developed by TRW and is now being built by Motorola. Devices are not yet commercially available. Table 13 shows a list of circuits now under development at Motorola[6] .

TABLE 13. EFL CIRCUIT TYPES

| Product | LSI Function | Speed | Approx. Devices on chip | Gate Equivalency* | LSI Chip Size ($mils^2$) | Product Designator |
|---|---|---|---|---|---|---|
| Microprocessor | | | | | | |
| 16 x 16 MPY | Multiplier | 975ns | 18,000 | 5,000 | 301 by 279 | XC5851 |
| 8 x 8 MPY | Multiplier | 390ns | 9,000 | 3,000 | 200 by 200 | XC5850 |
| PAU | Program Arithmetic Unit | 390ns | 3,000 | 1,000 | 230 by 291 | XC5810 |
| OAU | Operand Arithmetic Unit | 390ns | 3,000 | 1,000 | 315 by 355 | XC5820 |
| MPC | Microprogram Control | 50ns | 3,000 | 1,000 | 176 by 230 | XC5800 |
| CGA | Configurable Gate Array | | 5,000 | 1,500 | 270 by 279 | XC5830 |
| Special Circuit | | | | | | |
| BC | 64 Bit Correlator | 20MHz | 5,000 | 1,500 | 240 by 182 | XC5852 |

*Gate equivalency is derived by dividing the device count by 3. Three being the number chosen as components per 3 input nand gate.

The 3D, EFL structure[7] is shown in Figure 20 and is responsible for the favorable response of EFL in a radiation environment. The circuit uses a p-type substrate, vertical NPN and PNP transistors and buried N-type resistors. Since the p-type emitters in the PNP's, the p bases in the NPN's and the p region overlying the n-type resistors are heavily doped at the surface, the circuit is effectively immune to surface effects in $SiO_2$.

The transistors employ relatively wide bases which result in large transistor gain degradation under neutron bombardment. However, the design rules allow operation of the PNP's with gains of 2 and NPN's with gains of 3 before loss of logic margin occurs.

Various unhardened LSI EFL circuits have been radiation tested with the following results[5] :

(a) At $4 \times 10^7$ Rads(Si) under cobalt-60 exposure, the circuit remains functional with no loss in logic margin. At this level, the maximum frequency of operation is improved approximately 5 to 10%, as are the permissible power supply variations. The device betas have degraded to approximately 3 for PNP's and 13 for NPN's at this level. The ultimate failure dose has not been determined.

(b) Operational errors are made at a dose rate of $4 \times 10^8$ Rads(Si)/s. The level at which an operating circuit ceases to function during the radiation burst is about $5 \times 10^9$ Rads(Si)/s. The outputs are saturated at the ground state. At $9 \times 10^{10}$ Rads(Si)/s saturation at the ground state persists for approximately 1 $\mu$s after the radiation burst. Thereafter, it recovers to normal operation.

(c) At a cumulative fluence of $10^{14}$ N/cm$^2$, an operating circuit continues to operate without loss of logic margin. The maximum frequency of operation is observed to be approximately 30% higher than before exposure. At $2 \times 10^{15}$ N/cm$^2$ the circuit ceases to function, but operational recovery is achieved by thermal annealing at 150°C for 24 hours.

(d) Examination of the circuit structure may indicate the possibility of PNPN latch-up in the NPN transistor regions. However, this has not been reported.

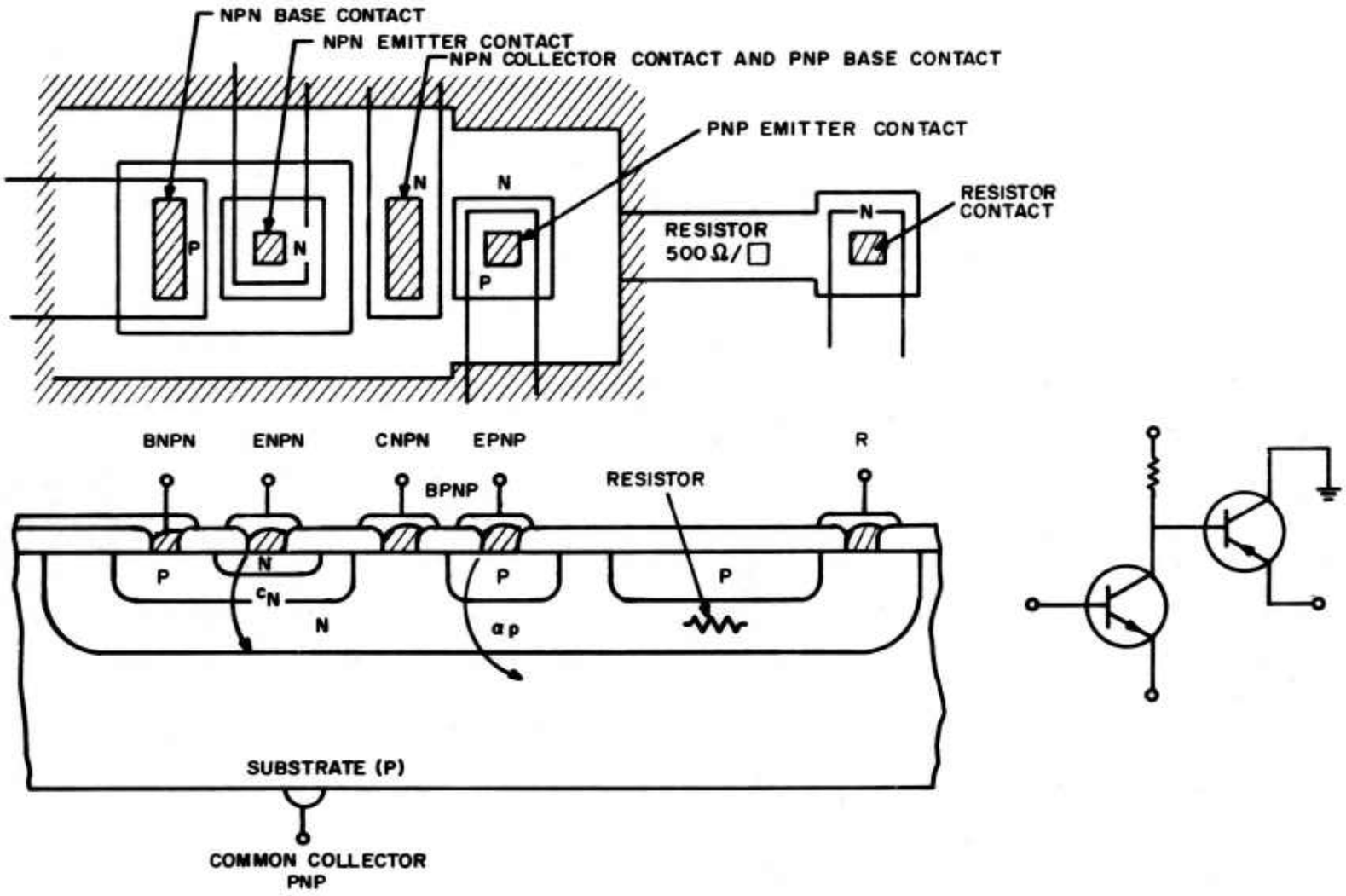



Figure 20. EFL Structure

(3) Integrated Injection Logic

IIL has a very attractive speed-power product, and it is a high density bipolar technology which allows digital and analog circuits to co-exist on the same chip. The structure and a simple logic gate are shown in Figure 21. The IIL gate has a single input and multiple outputs, in contrast to the multiple input, single output TTL.

Radiation effects in IIL tend to decrease the PNP alpha and also tend to invert the p-regions at the surface because of positive charge storage in the $SiO_2$. However, this latter effect is of minor importance since the impurity concentration in the p-region is rather high at the surface.

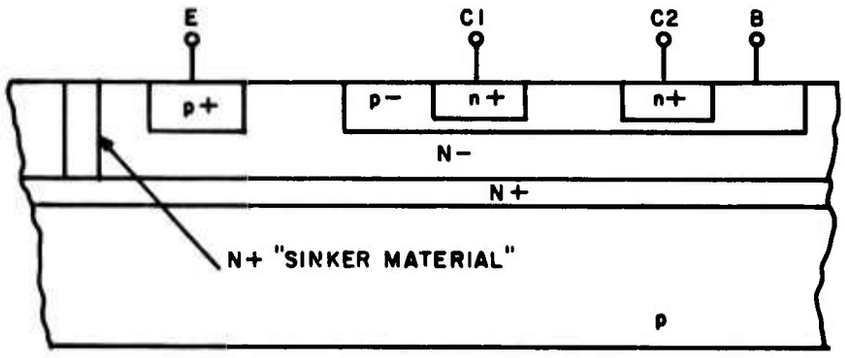
The generation of interface state under steady state ionizing radiation is a major source of radiation induced degradation in $I^2L$ circuits. The major effect is a reduction in the gain of the PNP transistors as shown in Figures 22 and 23 for both the PNP and NPN devices respectively.

$I^2L$ circuits are also degraded by neutron exposure in the same manner as other bipolar devices. Experiments carried out by Raymond et al.[8] have shown that failure occurs in the region of $3 \times 10^{13}$ N/cm$^2$ (on the basis of one device). However, severe degradation in $h_{FE}$ and "alpha" occurs in the region below $10^{12}$ N/cm$^2$ as shown in Table 14. Operation of the five-stage ring oscillator showed a reduction in frequency as a result of the transister gain degradation.

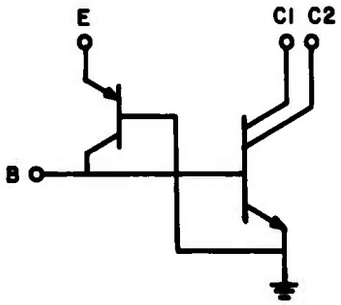
The results of the flash x-ray exposure are shown in Figure 24. First, it can be seen that latch-up did not occur. The maximum potential drop is one diode drop (approximately 0.8V) and there is no mechanism known that will support latch-up at such a low voltage. The data shows that the ring counter is affected by dose rates above $5 \times 10^8$ Rads(Si)/s. The counters simply increase frequency rather than skipping a beat. However, a partial recovery of the initial operating frequency is seen in the latter portions of each record shown in Figure 24.

(4) Transistor-Transistor Logic

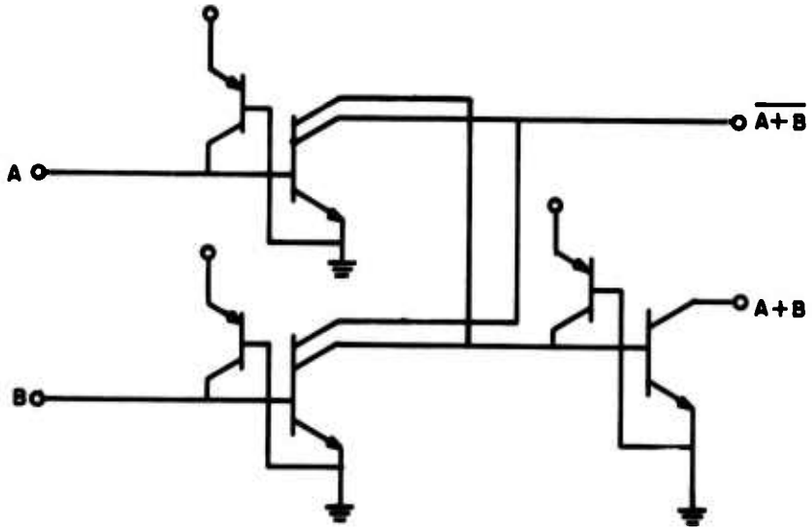
Hardened versions of many SSI TTL functions are available which incorporate gold doping, dielectric isolation, metal film resistors, low Z packaging techniques and photocompensation diodes in the circuit design. These circuits have been extensively tested and are known to operate at dose rates up to $10^9$ Rad(Si)/s. Many operate up to $10^{10}$ Rads(Si)/s) at full specified fan-outs of 10.[9] The most notable exception is the RSN 54L122 Monostable Multivibrator, for which the failure level is about $5 \times 10^7$ Rads(Si)/s. All




(a)
CROSS SECTION




(b)
GATE SCHEMATIC




(c)
IIL OR/NOR GATE

Figure 21. Integrated Injection Logic

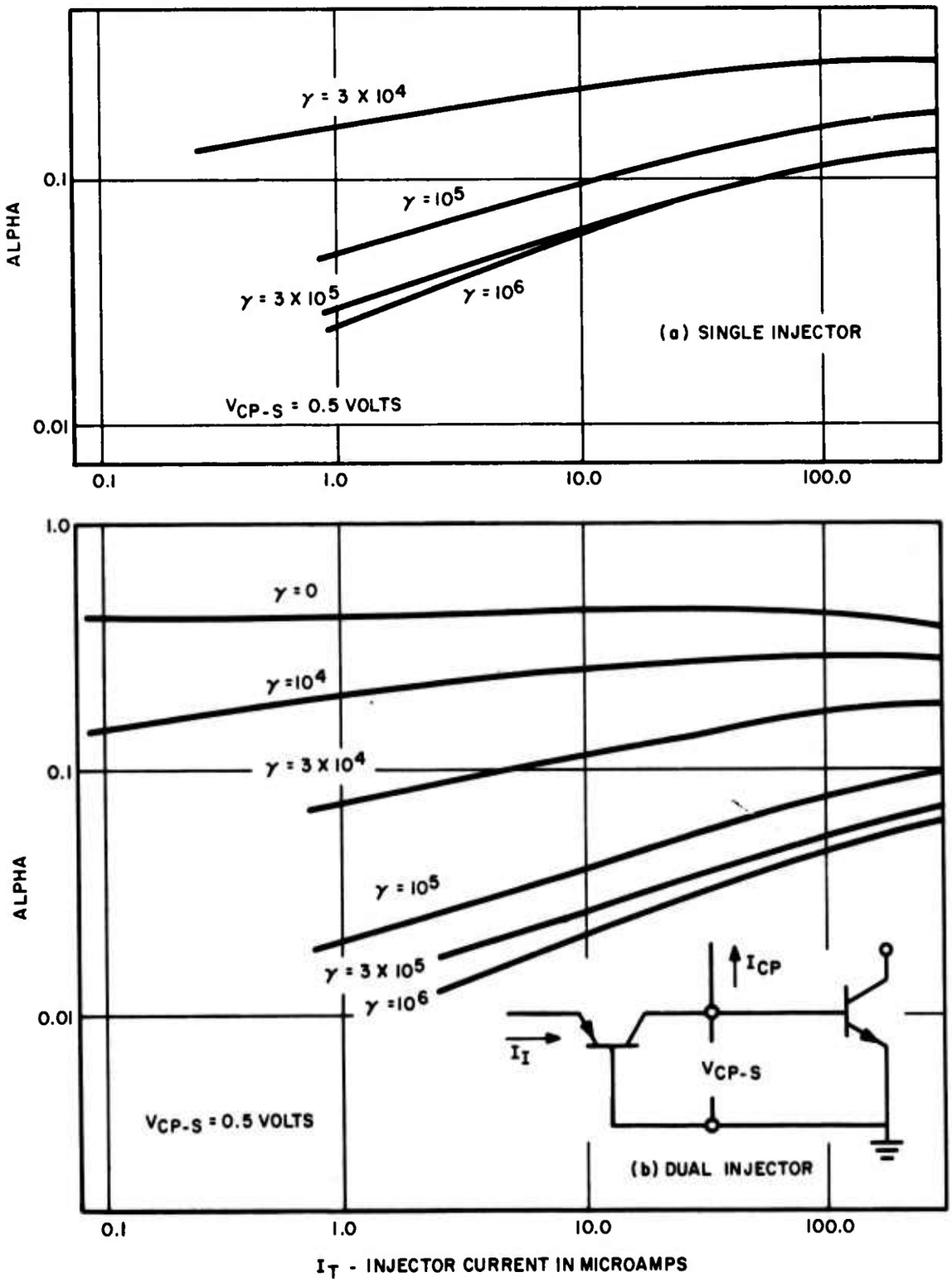



Figure 22. PNP Gain Degradation in IIL

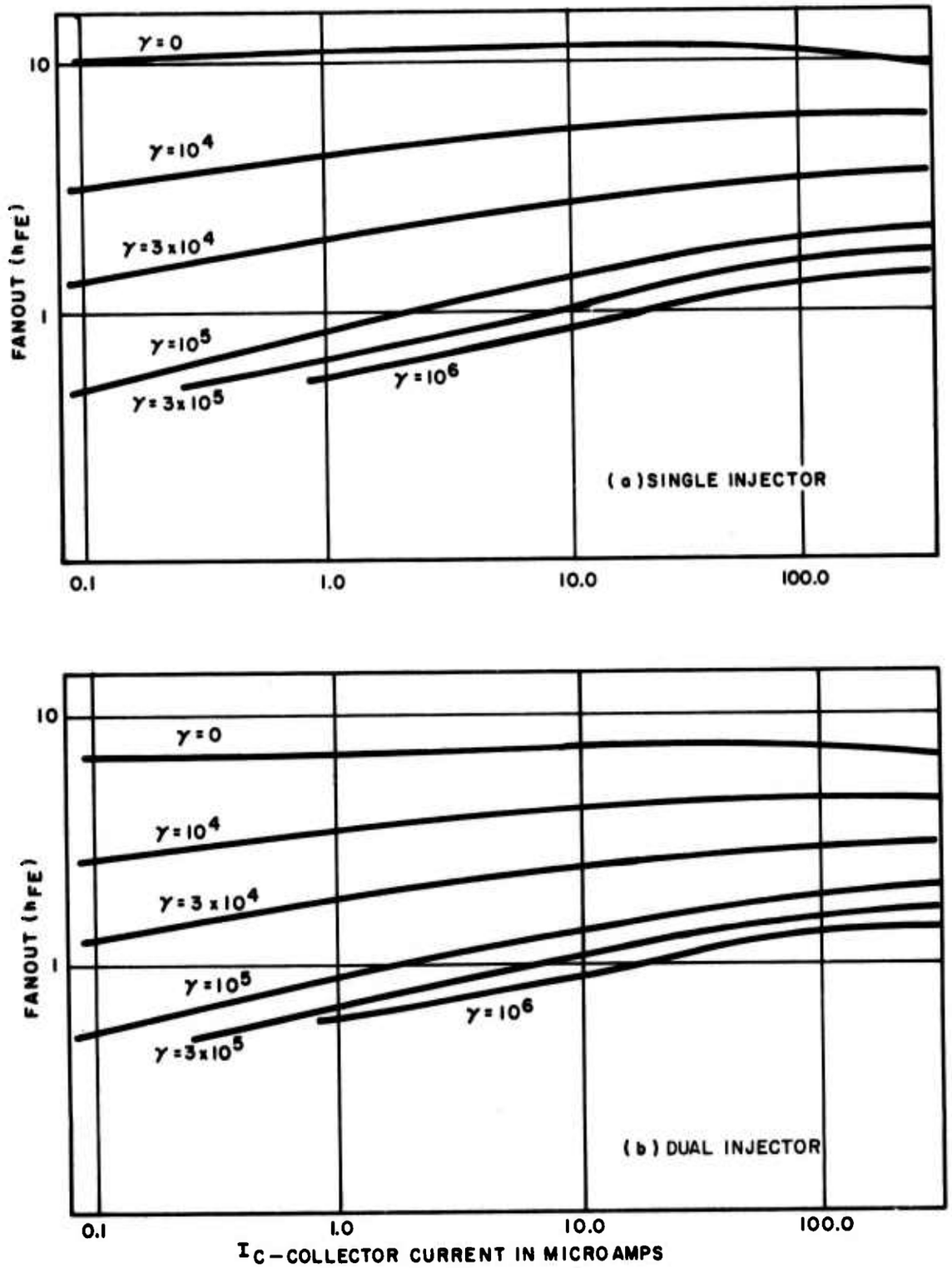



Figure 23. NPN Gain Degradation in IIL

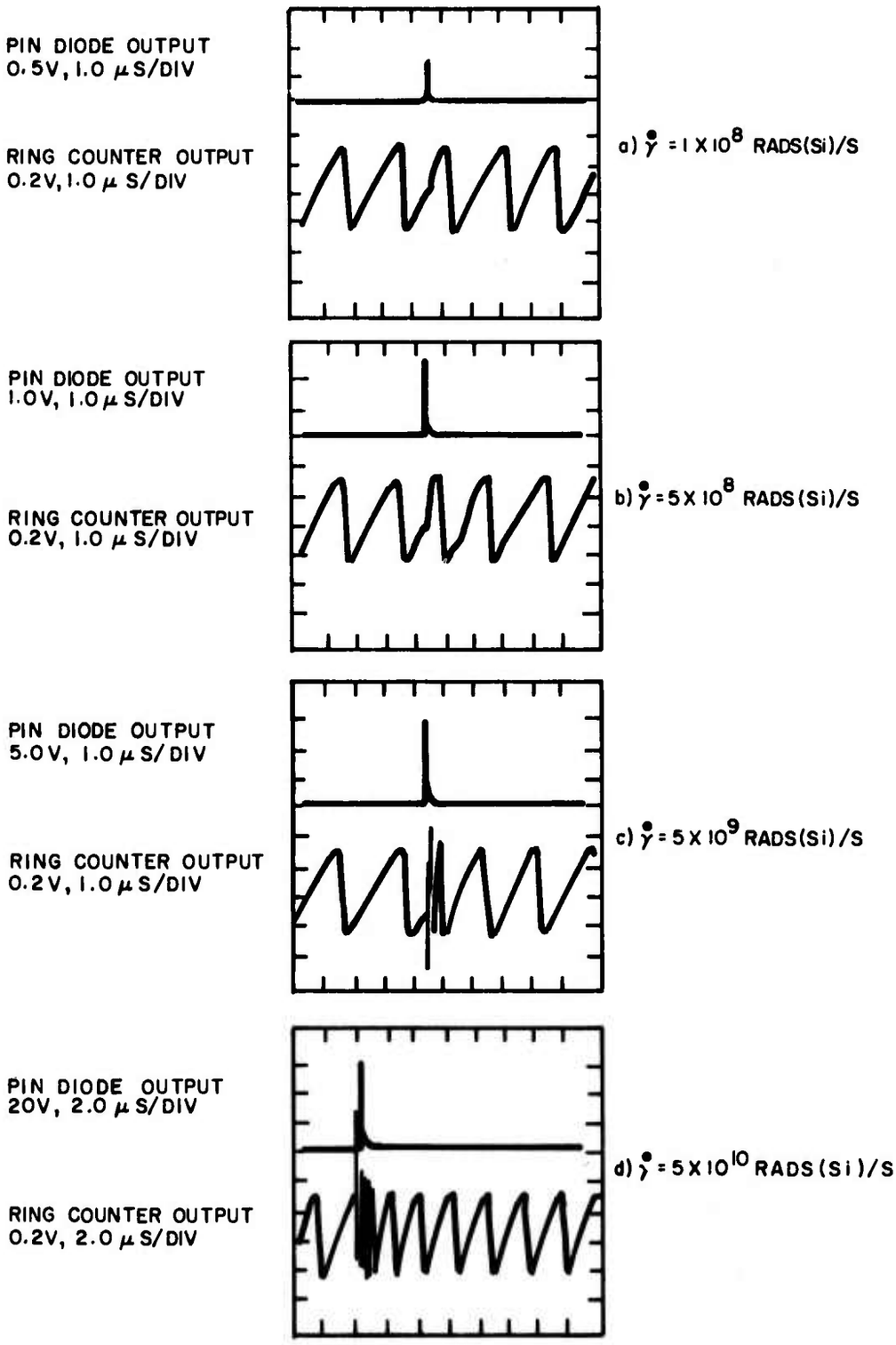



Figure 24. $I^2L$ Ring Counter Photoresponse

TABLE 14. $I^2L$ TEST CHIP NEUTRON-INDUCED DAMAGE

Transistor Common-Emitter Gain, $h_{FE}$

| Neutron Fluence $\Phi$, N/cm$^2$ | Sample #6 | | | Sample #7 | | |
|---|---|---|---|---|---|---|
| | Lateral PNP | Small NPN | Large NPN | Lateral PNP | Small NPN | Large NPN |
| 0 | 0.53 | 8.6 | 25 | 0.49 | 7.1 | 21.2 |
| $4.7 \times 10^{12}$ | 0.26 | 5.2 | 13.3 | 0.27 | 4.6 | 12.4 |
| $7.4 \times 10^{12}$ | 0.24 | 4.0 | 5.6 | 0.24 | 3.6 | 5.6 |

Ring Counter Frequency, $I_{EE}$ = 1 mA

| Sample | $\Phi$ = 0 | $\Phi = 4.7 \times 10^{12}$ N/cm$^2$ |
|---|---|---|
| #4 | 1.25 MHz | 0.74 MHz |
| #5 | 0.83 MHz | damaged in test |
| #6 | 1.09 MHz | damaged in test |
| #7 | 0.95 MHz | damaged in test |

devices operated satisfactorily at neutron fluences up to $1 \times 10^{14}$ N/cm$^2$ with acceptably small changes in output pulse levels and widths, with the exception of the RSN 54 L122, for which the pulse width decreased by 50-90%, and the RSN 54 L122 and the RSN 54 L71, for which the saturation voltage increased to about 1.0V. All circuits remained functional at total doses up to and above $10^7$ Rads(Si) in a steady state ionizing radiation environment with a 35-45% increase in rise time and no change in fall time. Other tests[10] using shallow planar geometries show significant degradation of the transistor gain at doses between $2 \times 10^5$ and $1 \times 10^6$, considerably lower than the levels given above.

Large scale integrated circuit radiation effect studies[8,10,11,12] are also underway. In these programs, the effort is aimed at ascertaining the radiation effects and failure levels for existing unhardened and hardened microcircuits now under development. The failure levels obtained by Raymond et al.[11] are shown in Table 15.

2. Field Effect Technologies

a. Damage Mechanisms

The primary radiation effects in FET's are well known and result from both the transient and steady state ionizing radiation. The effect of transient radiation pulses is the generation of photocurrents at each p-n junction in the circuit. The current may be reduced and the tolerance of the circuit may be increased by as much as 100 times, by placing the circuit on sapphire, one silicon island for each transistor. The result of this procedure is the reduction of the p-n junction area by about 100 times for a typical circuit.

The second equally important effect occurs under steady state ionizing radiation. Electron-hole pairs are created in the gate insulation and since, in $SiO_2$ more hole traps exist than electron traps, the oxide accumulates positive charge, moving the flat-band and thus the threshold voltage in the negative direction. An additional effect occurs in CMOS/SOS circuits. The sapphire also accumulates positive charge which tends to create a channel in the n-channel transistor at the silicon/sapphire interface which is manifested as an uncontrolled leakage current. In PMOS and PMNOS circuits, threshold shifts for identical transistors tend to be somewhat higher in magnitude, so that circuits that will tolerate $3 \times 10^6$ Rads(Si) on bulk silicon will tolerate only $3 \times 10^5$ Rads(Si)[12] when implemented on sapphire. The decrease in total dose hardness also seems to be true for CMOS/SOS [13]. The threshold shift is strongly dependent upon the bias applied during exposure.

# TABLE 15. MSI/LSI BIPOLAR VULNERABILITY

| | Signetics Memory Systems Read Only Memory SMS 8228 | Harris Radiation Hardened Adder H 6BA | Fairchild μA 770 Radiation Hardened Operational Amplifier | Texas Instr. SN54S181 Arithmetic Logic Unit | Texas Instr. Low Power Schottky-Clamped TTL/MSI | | |
|---|---|---|---|---|---|---|---|
| | | | | | SN74LS139 | SN74LS155 | SN74LS194 |
| Pulsed Ionizing Radiation | | | | | | | |
| a) Narrow pulse transient failure level $T_p$ = 30 ns, 2 MeV FXR | 2-3 x $10^7$ Rads(Si)/s | 3-30 x $10^8$ Rads(Si)/s | Peak-to-peak output voltage level at 20 to 30 V for dose rates above 5 x $10^9$ Rads(Si)/s, below which there is an approximately linear relationship down to .2 to .5 V at $10^7$ Rads(Si)/s | 1.5-4 x $10^8$ Rads(Si)/s | 5-10 x $10^8$ Rads(Si)/s | 1.5-3 x $10^8$ Rads(Si)/s | 1.5-2 x $10^8$ Rads(Si)/s |
| b) Wide pulse transient failure level $T_p$ = 4 microseconds | ~2 x $10^6$ Rads(Si)/s | 2-20 x $10^8$ Rads(Si)/s | | ~2 x $10^7$ Rads(Si)/s | ~3 x $10^7$ Rads(Si)/s | ~3.5 x $10^7$ Rads(Si)/s | ~1.5 x $10^7$ Rads(Si)/s |
| c) Permanent damage failure level $T_p$ = 30 ns, 2 MeV FXR | ~3 x $10^{13}$ Rads(Si)/s | ~6 x $10^{12}$ Rads(Si)/s | 6-30 x $10^{12}$ Rads(Si)/s | | | | |
| Neutron Permanent Damage | | | | | | | |
| a) Mean failure level at room temperature | 4.9 x $10^{14}$ N/cm$^2$ (1 MeV equiv) | | | ~3 x $10^{14}$ N/cm$^2$ (1 MeV equiv) | 7.8-12 x $10^{14}$ N/cm$^2$ (1 MeV equiv) | ~12 x $10^{14}$ N/cm$^2$ (1 MeV equiv) | ~5.7-7.8 x $10^{14}$ N/cm$^2$ (1 MeV equiv) |
| b) Concomitant gamma dose | ~ $10^6$ Rads(Si) | | | ~$10^6$ Rads(Si) | ~2 x $10^6$ Rads(Si) | ~2 x $10^6$ Rads(Si) | ~2 x $10^6$ Rads(Si) |
| c) Observed failure level range | < 2 to 6.6 x $10^{14}$ N/cm$^2$ | | | 2-10 x $10^{14}$ N/cm$^2$ Depending on fan out specification | | | |
| d) Mean failure level case temp at 0°F | 3.4 x $10^{14}$ N/cm$^2$ (1 MeV equiv) | | | | | | |
| Total Ionizing Radiation Total dose failure level | | > $10^7$ Rads(Si) | | | | | |

The procedures for hardening MOS devices in general are listed in Table 16. Sapphire will drastically reduce junction area and increase transient hardness. Various processing techniques can be used to decrease threshold voltage sensitivity to total dose. Design tradeoff between speed and power can also be made to increase hardness.

TABLE 16. MOS HARDENING TECHNIQUES

- Use sapphire to reduce p-n junction area and transient photocurrents
- Use an "ultra pure" gate oxide with aluminum ion implant in the n channel insulator followed by a high temperature anneal
- Use aluminum oxide for the gate insulator to provide electron as well as hole traps. This technique seems to produce unstable devices that exhibit threshold voltage changes with applied voltage
- Use a chrome doped gate insulator
- Trade off speed and power for radiation hardness

b. Logic Family Performance

(1) CMOS and CMOS/SOS

In one standard CMOS process, both the n and p channel transistors are enhancement mode devices. Under radiation, the n-channel device can become a depletion mode device, affecting the proper logical operation of the circuit.

Figure 25 is an example of the effect of total dose radiation on CMOS with an $Al_2O_3$ gate oxide[14]. Similar transfer curves for the other gate insulator have been obtained[14].

(2) PMOS and PMOS/SOS

PMOS circuits are adversely affected by steady state ionizing radiation because of the shift in threshold due to oxide charging and interface state creation. In order to reduce the p-n junction photocurrents, circuits may be fabricated on sapphire. This brings in an additional possible

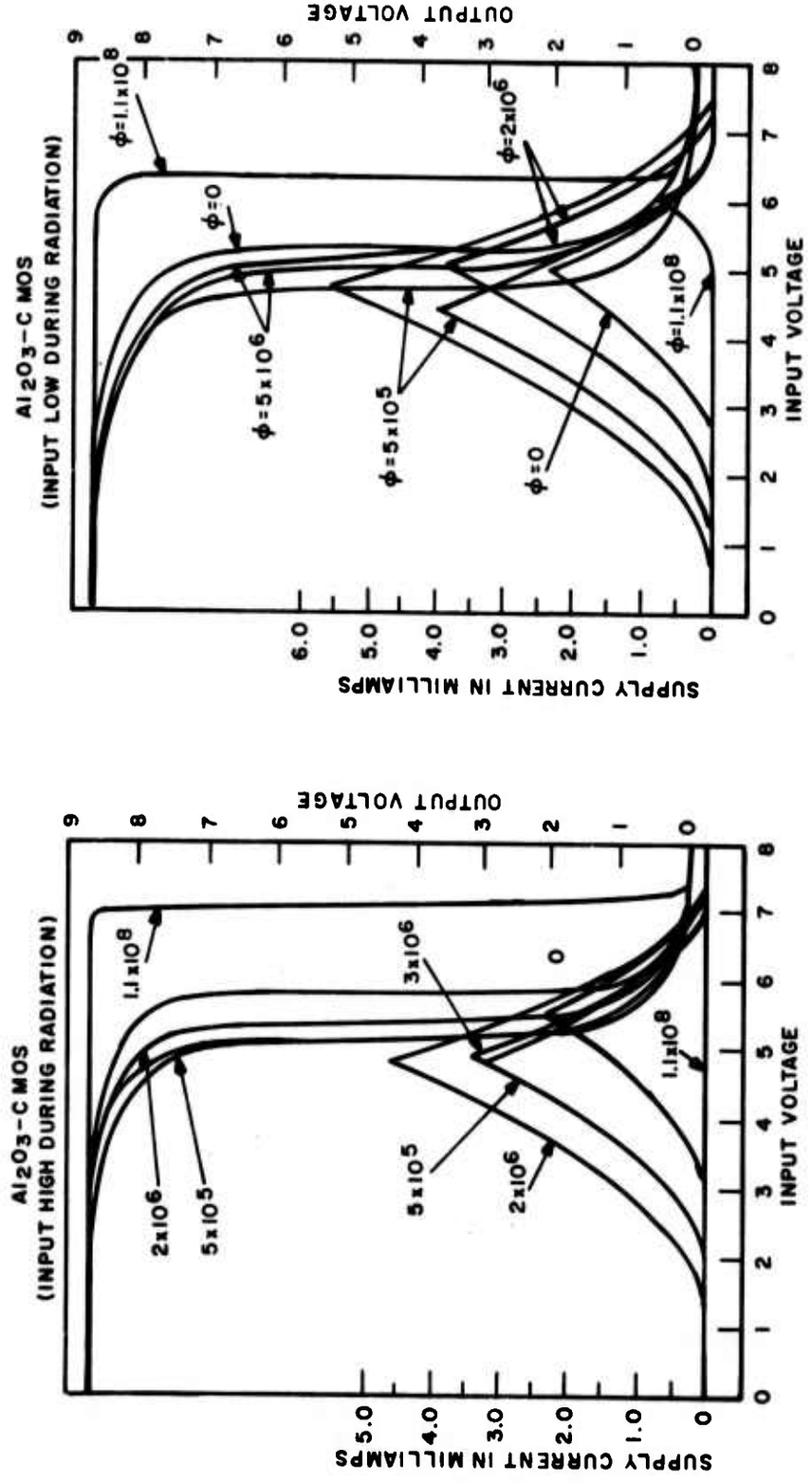



Figure 25. CMOS Total Dose Performance

failure mechanism, source-to-drain leakage current. Reference data[12] shows the leakage current in a typical PMOS device on sapphire is initially low and that it remains low (in some cases actually decreases) at doses up to $3 \times 10^6$ Rads(Si). Figure 26 shows the threshold voltage of a typical transistor, indicating a 2V shift at $3 \times 10^6$ Rads(Si). These are typical values for hardened devices using the "Ultra Pure" Aubuchon oxide, a process that can be used even for devices on sapphire.

Microcircuits have also been tested[12] using dynamic PMOS circuits on sapphire, as well as bulk PMOS circuits. The observed failure levels are shown in Table 12.

(3) NMOS and MOS/SOS

NMOS devices can be operated in two modes - either in their enhancement mode, such as CMOS circuits, with positive gate threshold voltages or in their depletion mode with attendant negative threshold voltages as used in modern high speed NMOS microcircuits. No radiation test was unearthed during this study for high speed NMOS devices. The hypothesis, however, is that since the thresholds are negative and since they will change in the negative direction as in PMOS devices, such devices should behave in much the same way as PMOS devices and should become as hardened using the pure oxide approach.

## C. HYBRID PACKAGING

The design and specification of multichip hybrid packages requires many tradeoffs. A detailed treatment of the many considerations are given in the references. Some general comments and a few examples are given in the following paragraphs, along with a description of a hybrid packaged microprocessor system.

### 1. General Comments

The methods of bonding the chip to the substrate may be grouped into face-up and face-down classes. Chip and wire methods are accepted, industry wide procedures, while beam leads and bumps represent recent bonding techniques. Face-up mounting allows higher chip power dissipation due to the lower thermal resistance to the package, and the chip area is usually smaller with this bonding technique.

Interconnection wiring within the package may be accomplished with single or double-sided substrates, as well as multilayer arrangements. Single layer limitations are interconnection area and the availability of cross-overs. Multilayer interconnection requires additional material and more complicated

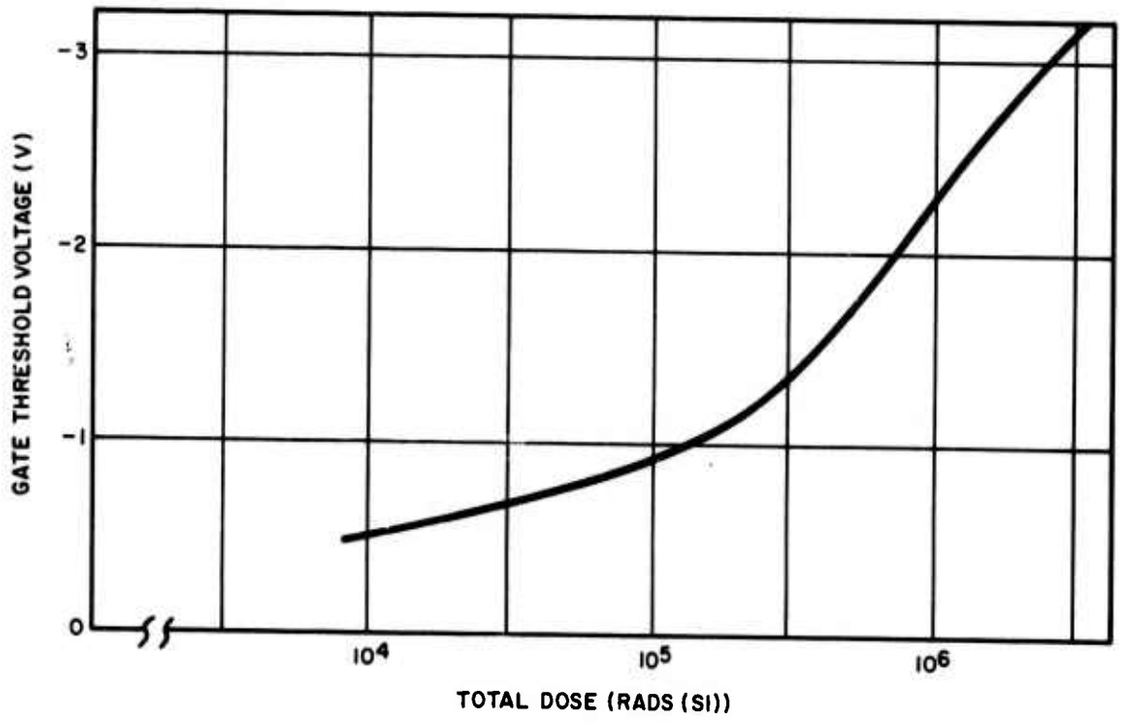



Figure 26. PMOS Threshold Voltage Change

processing. Substrate design tradeoffs between conductor and dielectric material, conductor width and thickness will affect electrical performance such as inductive and capacitive coupling, characteristic line impedance, crosstalk and propagation time.

2. Examples

The following examples of multichip packaging were obtained from trade literature and represent the kind of systems applicable to this contract.

a. CMOS Memory for MJS Spacecraft

Algorex Data Corporation is supplying a thick film hybrid memory circuit to the Jet Propulsion Laboratory for use on the Mariner spacecraft, which will be launched in 1977 to rendezvous with Jupiter and Saturn. The memory will consist of four, 3 x 3 inch, four-layer hybrid circuits, each containing 2048 16-bit words. Each hybrid contains 140 chips: 128 CMOS memory chips and 12 driver chips. The memory chips are RCA CD4061 static RAMS with a 256 x 1 organization. The four hybrids do the job of two 7 x 14 inch multilayer boards, and the cost, in the low hundred-thousand-dollar range, is reportedly half the conventional approach.

b. Missile CPU

The same company is also building a 16-bit central processing unit in hybrid form for a missile system. The CPU consists of 75 digital IC's and 25 discrete active and passive chips. The ceramic substrate is two inches square and contains 10 layers. The high speed processor operates with a 20 MHz clock time.

c. Honeywell CPU

Honeywell is using hybrid circuits in its MOD/LSI line of aerospace computers. One of the computers, the MOD/LSI-2, was chosen for the Navy's advanced tactical inertial guidance system (ATIGS) for small missiles. In what may be the same hybrid described above, approximately 75 IC's and 25 discretes are combined on a two inch square substrate and housed in a 120 lead package. This 16 bit processor uses TTL and low-power Schottky TTL technologies. The computer can perform 250,000 - 400,000 operations per second, depending upon the circuit implementation, memory technology and instruction mix.

d. Microprocessor System

Hybrid packaging is used by Teledyne Semiconductor to form a microprocessor system. A 120 lead package contains 40 IC's and 10 discrete chips. Flying leads connect the chips to the multilayer substrate. The package photograph appears to contain eight memory arrays and eight processor chips. The Teledyne TDY-52B hybrid processor is described in detail in the following paragraphs.

e. Hybrid Microprocessor System

Teledyne Systems Company is making microprocessor systems using multichip hybrid packaging. The package characteristics of the TDY-52B are given below:

43 IC's, 22 Resistors, 6 Caps, 2 Transistors, 5 Diodes

120 Lead package, 2" x 2" x 0.2"

Multilayer Substrate

Chip and Wire Interconnect

Hermetically Sealed

Commercial Components: 0° to 70°C

Typical Power Dissipation - 5.4 W

The integrated circuits are commercial temperature range TTL, and low power Schottky TTL SSI and MSI, and PMOS slices for the CPU and CROM. The integrated circuits have approximately 575 bonding pads, so that 1400 bonding operations are required for chip-substrate and substrate-package interconnection. Only 25 to 33% of the substrate is active chip area.

The characteristics of the 16-bit processor are shown in Table 17. The processor uses the National Semiconductor IMP-16 four bit RALU slice and a 100 word by 23-bit ROM which implements the standard instruction set. A second CROM is required for the extended instruction set which includes multiply and divide and double precision add and subtract.

A block diagram of the TDY-52B is shown in Figure 27. The Clock Generator develops the necessary clock phases for the MOS clock drivers and CPU timing signals. There are eight phases in the basic 1.4 $\mu$s microcycle. TTL buffers are included in the package to buffer the MOS RALU signals. The RALU devices are controlled by a four bit, time multiplexed bus from the CROM. During normal program execution, the user's macro instruction is

TABLE 17. TELEDYNE HYBRID MICROPROCESSOR CHARACTERISTICS

| | |
|---|---|
| Organization | Parallel 16 bit, Microprogrammed<br>General Register/Accumulator |
| Registers | 4 General Purpose<br>4 Special Purpose<br>16 Level Stack |
| Instructions | |
| Basic Set | 43 |
| Extended Arithmetic | 17 Double Precision, Multiply, Divide |
| Input/Output | 16 Bit Peripheral Data In Bus<br>16 Bit Memory Data In Bus<br>16 Bit Data Out Bus<br>16 Bit Address Out Bus<br>2 Interrupt Lines<br>4 Jump Condition Flag Inputs |
| Instruction Times | |
| Reg/Reg Add | 4.6 $\mu$s |
| Input/Output | 10 $\mu$s |
| Power Requirements | +5 V dc at 0.6A<br>-12 V dc at 0.2A |
| Price (50-500 units) | $1295.00 |

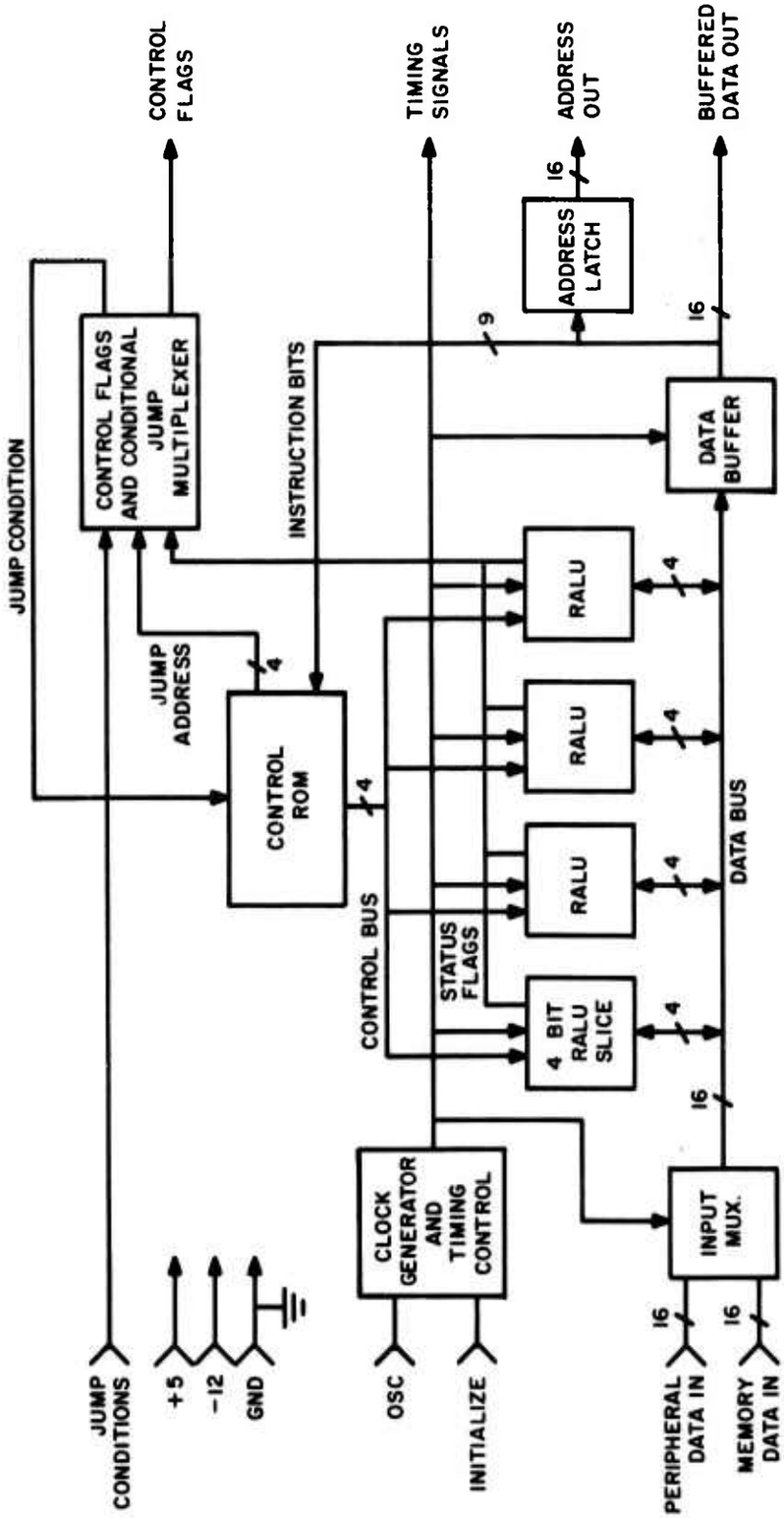



Figure 27. TDY-52B Block Diagram

fetched from external memory. The instruction is decoded, directing the control sequence to an entry point in the microprogram. When the microprogram execution is complete, the next macro instruction is fetched. Macro program execution can be suspended if an interrupt condition exists. Microprogram paths are affected by the RALU status flags, and user supplied jump conditions.

The chip types contained in the TDY-52B are listed in Table 18. One impressive observation is the large number of SSI support chips required. This result is typical for other 16 bit systems configured from competing 4 bit slices. Smaller systems could be configured at the expense of some of the powerful features that the processor has. The processor configuration (the number of SSI parts and package pins) is determined by the application requirements.

The Teledyne TDY-52B is an example of the industry wide capability in multichip hybrid packaging. The capability can be extended to any chip technology with the appropriate design constraints.

The state-of-the-art in hybrid technology and device processing is exemplified by the Trident Program. A blend of technologies which have not been combined before are being used to fabricate hybrid circuits for use in the command sequencer, interlocks and flight control electronics. The three technologies, beam lead sealed junction, dielectric isolation and low power Schottky TTL, produce low yield military devices when used singularly, and result in lower yields when combined. Although some circuit and process improvements have been made, some substitution of junction isolated circuits will probably be made for the production program.

TABLE 18. TDY-52B CHIP TYPES

| Type | Quantity | Function | Integration | Pins |
|---|---|---|---|---|
| 74LS00 | 2 | Quad NAND | SSI | 14 |
| 74LS02 | 1 | Quad NOR | SSI | 14 |
| 74LS04 | 1 | Hex Inverter | SSI | 14 |
| 74LS08 | 8 | Quad Buffer | SSI | 14 |
| 74LS11 | 1 | Triple NAND | SSI | 14 |
| 74LS32 | 1 | Quad OR | SSI | 14 |
| 74LS54 | 1 | AND-OR-INV | SSI | 13 |
| 74LS74 | 5 | Dual D F/F | SSI | 14 |
| MH0026 | 2 | MOS Drvrs. | SSI | 6 |
| 8095 | 1 | Hex Buffer | SSI | 16 |
| 74195 | 1 | 4 Bit Reg. | MSI | 16 |
| 8121 | 2 | 8 Chan Mux | MSI | 16 |
| 85L51 | 4 | Quad D F/F | MSI | 16 |
| 9322 | 1 | Quad Mux. | MSI | 16 |
| 9334 | 2 | 8 Bit Reg. | MSI | 16 |
| 81L23 | 4 | Quad Mux. | MSI | 16 |
| MM5750 | 4 | RALU | LSI | 24 |
| MM5751 | 2 | CROM | LSI | 24 |
| | 43 | | | |

Integration:

14% LSI

33% MSI

53% SSI

Interconnections:

575 IC Pads x 2 = 1150

120 Pkg Pins x 2 = 240

Total 1390

# SECTION IV

# ARCHITECTURE STUDIES

## A. MICROPROCESSOR STATE-OF-THE-ART

In 1971 Intel introduced one of the first single-chip microcomputers - the 4004. The initial devices were an outgrowth of the commercial calculator developments and were generally four-bit devices with decimal arithmetic capability. The second generation devices featured a longer eight-bit word and the technology changed from the slower p-channel to the faster n-channel MOS process. Present generation devices show evolution along two architectural lines. More computing power can be achieved either by building denser, single chip Central Processing Units (CPU) or by structuring microcomputers out of multiple byte slice elements. Single chip, 16-bit devices, such as the National PACE, General Instrument CP-1600, and General Automation LSI-16, have been fabricated, showing one trend in the direction of single-chip, stand-alone microcomputer systems. The other approach offers two-bit and four-bit byte slice elements from National, Intel, Monolithic Memories and Texas Instruments. The slice devices are predominantly bipolar technologies (Schottky and Integrated Injection Logic) for speed reasons, while the MOS processes are retained for high density in the 16-bit processors.

Of the two choices, the byte slice architecture seems to be more compatible with military mission objectives. It provides a wide temperature range of operation and meets the desired computing requirements of a broad spectrum of applications. It also provides a means of building a range of microcomputer architectural structures. The byte slice architecture offers the designer a standard building block which allows him to tailor the machine organization to his own needs.

The characteristics of mid-1975 microprocessors are given in Table 19. The characteristics are loosely grouped into three areas: credibility-risk, hardware design and software design. In addition to the tabular data, the following paragraphs discuss several of the microprocessors in detail.

# TABLE 19. MICROPROCESSOR CHARACTERISTICS

| | CREDIBILITY-RISK | | | | | | | HARDWARE DESIGN | | | | | | | ARCHITECTURE | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Availability | Second Source | Price $/1000 | Family Parts | Process | Chip Size | Hardware Support / Software Support | Cycle Time, μs | A+B C, μs | Power/Chip, Watts | Power/CPU, Watts | Voltages | Package Pins | Temperature Range | Word Length | Instruction Set | Microprogramable | Registers | Stack Size | Memory Size | Address Modes | Interrupt |
| Rockwell PPS-4 | Y | N | 22 | H | P | | H H | 5 | 2.5 | .25 | .25 | -17 | 42 | 0-70 -35+125 | 4 | L | N | 2 | 2-12 | 4K | | |
| Intel 4004 | Y | N | <30 | H | P | 117x159 | H H | 10.8 | | .45 | .45 | -15 | 16 | 0-70 | 4 | L | N | 24 | 3-12 | 4K | | |
| Intel 4040 | Y | N | <30 | H | P | 118x163 | H H | 8 | | .55 | .55 | -15 | 24 | 0-70 | 4 | L | N | 24 | 7-12 | 4K | | L |
| Intel 8008-1 | Y | | 75 | H | P | 124x173 | H H | 12.5 | 22.5 | .42 | .42 | +5,-9 | 18 | 0-70 | 8 | M | N | 7-8 | 7-14 | 16K | | M |
| Intel 8080 | Y | AMD?, TI | | H | N | 164x191 | H H | 2 | | .78 | .78 | +12, +5 -5 | 40 | 0-70 | 8 | M | N | 6-16 | RAM | 64K | 4 | M |
| Fairchild F8 | ? | Mostek | | H | N | | H H | 2 | | <1 | <1 | +12, +5 | 40 | 0-70 | 8 | M+ | N | 64-8 | RAM | 64K | | L |
| Mostek MK5065 | ? | | 50 | - | P | >200x200 | L M | 3 | | .7 | .7 | +5, -5, -12 | 40 | 0-70 | 8 | L | N | 10 | RAM | 32K | 4 | M |
| Motorola 6800 | ? | AMI | 30 | H | N | 210x210 | M M | 2 | 13 | 1.2 | 1.2 | +5 | 40 | 0-70 | 8 | M+ | N | 3-8 3-16 | RAM | 64K | 5 | M |
| Rockwell PPS-8 | ? | | 34 | H | P | 175x180 | H H | 12 | 50 | .3 | .3 | -17 | 42 | 0-70 -35+125 | 8 | M | N | 4 | RAM | 16K | | M |
| RCA COSMAC | ? | AMS? | 50? | L | C | | H H | 3 | >50 | 0.1 | 0.1 | 5 | 28 40 | 0-70 MIL? | 8 | M | N | 16-16 | | 64K | | |
| Signetics 2650 | N | | 200(100) | L | N | >200x200 | - - | 10 | ≈24 | 0.3 | 0.3 | 5 | 40 | 0-70 | 8 | M | N | 7-8 | 7-15 | 32K | 8 | M |
| West. Dig. 1600 | N | | <200 | - | N | 184x209 | - - | .3 | 1.1 x PDP-11 | | | +12, +5, -5 | 40 | 0-70 | 8 | H | Y | 26-8 | | 64K | | |
| SMS | ? | | 385(100) | H | B | Hybrid | H H | .3 | | | | 5 | NA | 0-70 | 8 | L | Y | 11 | | 8K | | |
| GE/SSS CRD8 | N | | | M | SOS | | H H | .2 | | .01 | <.1 | 5 | 40 | 0-70 MIL? | 8 | M+ | Y | 8x16 | RAM | 64K | 7 | L |
| Intersil 6100 | ? | | | 0 | C | | H H | .5 | 20.5 | .01 | .01 | 5 | 40 | 0-70 -55+125 | 12 | M+ | N | 6-12 | | 4K | 5 | M |
| Toshiba TLCS12 | ? | | 73 | H | P | >200x200 | H H | | 99 | 1.0 | 2.0 | +5,-5 | 16-42 | 0-70 -40+85 | 12 | H | N | 10 | RAM | 4K | | H |
| National IMP 4/8/16 | Y | | 70, 73, 140 | M | P | <200x200 | H H | 1.4 | 25.6 | .7 | - | -5,-12 | 22 24 | 0-70 | 4N | H | Y | 8 | 16 | 64K | 7 | H |
| Intel 3000 | Y | | 3001-22 3002-13 | M | BS | | - L | .15 | | .725 | 3.75 | +5 | 40 28 | 0-70 | 2N | NA | Y | 11 | - | - | - | - |
| MMI 5701 | Y | | 95(100) | - | BS | 160x145 | - L | .2 | | 1.25 | 5.0 | +5 | 40 | 0-70 MIL? | 4N | NA | Y | 17 | - | - | - | - |
| TI 0400 | Y | | <90 | L | BI | | L L | <1.0 | | .16 | <1 | +5 | 40 | 0-70 MIL? | 4N | - | Y | 10 | | | | |
| National PACE | Y | | 120 | - | P | 235x235 | H H | 2.0 | 26 | | | | 40 | 0-70 | 16 | H- | N | | 10-16 | 64K | | |
| GI CP 1600 | Y | EM+M | 50(100K) | L | N | 200x200 | M H | 2.4 | 3.2 | | | +12, +5, -3 | 40 | | 16 | M | N | 8-16 | RAM | 64K | | M |
| GA LS1-16 | ? | | | | SOS | 200x200 | H | 1.8 | | | | | | | 16 | | Y | | | 32K | | |

Y = Yes
N = No

H = High
M = Medium
L = Low

N = NMOS
P = PMOS
C = CMOS

B = Bipolar
BS = Bipolar Schottky
BI = Bipolar IIL

SOS = Silicon on Sapphire

1. Intel 8080

a. Hardware Considerations

The Intel 8080 is an eight-bit parallel CPU packaged in a 40-pin Dual In-Line Package (DIP). It requires multiple power supplies (±5V, +12V) and a two-phase, high-level clock. In normal operation the chip dissipates 1.3 W. The NMOS device is also being supplied by AMD and TI. A block diagram of the chip is shown in Figure 28.

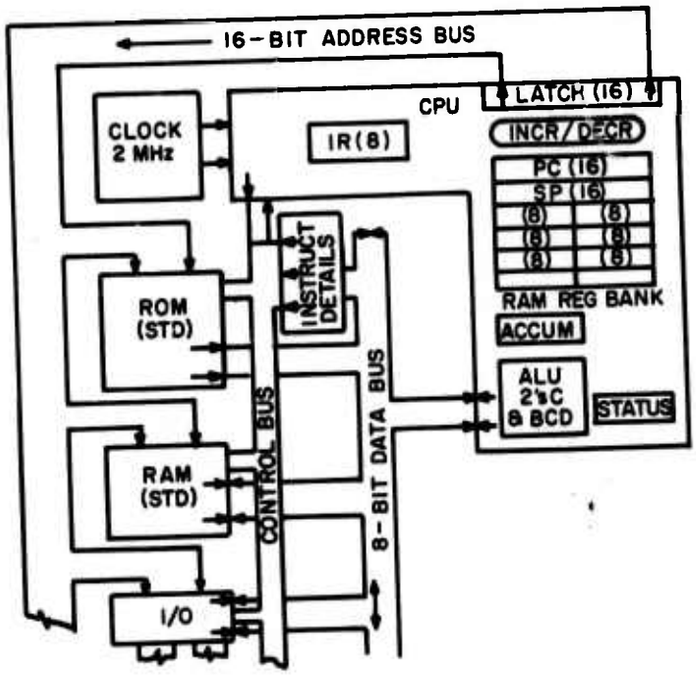



Figure 28. Intel 8080 Block Diagram

Architecturally, the 8080 has a three-register 16-bit file and an accumulator. For compatibility with the Intel 8008, many instructions treat these as seven separate eight-bit registers. The 16-bit stack pointer is used to place all return addresses in RAM. Of course, this means the program counter has to be assigned a unique on-chip register. Unlike the 8008, the 8080 has a separate 16-bit address bus (maximum of 65K of memory). The stack, in Random Access Memory (RAM) can also be used to store data. It is of unlimited depth (to the limits of storage), although execution times for stack-referencing call instructions are slightly longer.

The 8080 operates at a clock frequency of 2 MHz. The time required for a register-to-register add is 2 $\mu$s. The register load time for an instruction is 2.5 $\mu$s and the register-to-memory add time is 3.5 $\mu$s.

b. Hardware Support

Intel provides an extensive spectrum of possibilities from chips to chip sets to boards up to minicomputer (Intellec 8 MOD 80) consoles.

c. Software

The data and instruction word size is eight bits in length with 16-bits for addressing. By design there is major software compatibility between the 8080 and 8008. The 8080 has a total of 78 instructions. The 8080 permits any of the three main registers to hold and output an address when using register-indirect addressing. Multiple indexing can be accomplished with these, but it takes additional steps compared to the classical index register concept. The push down stack can be used to hold status bits and important register contents for interrupt servicing. The data manipulation instructions include arithmetic, logic, BCD arithmetic, and double precision operations. All general purpose registers can be incremented or decremented. The use of the stack pointer (SP) allows unlimited subroutine nesting.

d. Software Support

Intel has made a heavy investment in developing a high level language for this machine (PL/M). Macro and resident assemblers are available from Intel and General Electric. Intel also provides an instruction simulator and cross assembler.

e. System Considerations

Bus controls allow for Direct Memory Access (DMA). The 8080 has multiple interrupt capability which is software enabled or disabled. However, it cannot be microprogrammed. The Input/Output (I/O) is accomplished on the same tristate data bus and is limited to 256 ports.

f. Comments

The 8080 has some potential disadvantages:

(1) The lack of indexed addressing is serious in some applications.

(2) The 8080 needs three power supplies.

(3) There is no 2's complement arithmetic for signed numbers.

However, the 8080 is a machine which has set the standard for second-generation performance in eight-bit microprocessors. The PL/M language developed by Intel is the industry trend to support software standardization.

2. Motorola 6800

a. Hardware Considerations

The Motorola 6800 is an integrated chip set with some special parts for input/output. The eight-bit CPU is packaged in a 40-pin DIP. It requires a single +5 V supply with a two-phase, 5 V clock. Normal operation requires a maximum of 1.2 W of power. Figure 29 is a block diagram.

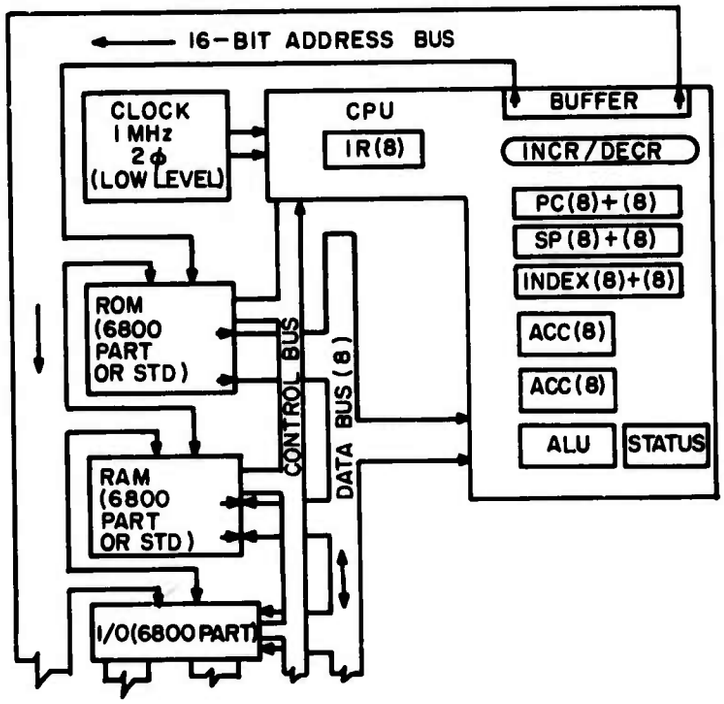



Figure 29. Motorola 6800 Block Diagram

The CPU itself is organized as a conventional computer with an eight-bit data/instruction path. Instructions execute in 2 to 12 $\mu$s. There are two accumulators and a 16-bit index register. Return addresses are stored in RAM and that stack is referenced via a pointer in the CPU. There are 16 bits of address available; however, abbreviated addresses of eight bits can be directed to refer to the first 256 bytes of main memory.

The 6800 operates with a 1MHz two-phase clock. A register to register add is accomplished in 2 $\mu$s. A register to memory add time is 4 to 6 $\mu$s depending on address modification.

b. Hardware Support

Motorola provides special LSI integrated circuits to interface with the CPU chip. Specifically they provide a 1024 x 8 bit Read Only Memory (ROM), a 128 x 8 bit RAM, a bi-directional Peripheral Interface Adapter (PIA), and an Asynchronous Communications Interface Adapter. Also various levels of printed circuit board and console aids are being developed.

c. Software

The data and instruction word size is eight bits in length with a 16-bit address. There are 72 instructions available. Programming the 6800 is relatively easy, but there are disadvantages. There are, for example, no direct paths from the two accumulators to the index register. To compute an index requires storing the 16-bit value into RAM and then reading it into the index register. The choices of various addressing modes are somewhat arbitrary because only certain instructions admit certain modes; it is the programmer's responsibility to remember which addressing forms are valid for each instruction. The use of the stack pointer allows almost unlimited subroutine nesting. The instructions are arithmetic and logical which take advantage of the two accumulators. The 6800 has instructions for storing the status register. Relative addressing allows data relocation. The programmer can reach the first 256 locations of memory with "short" instructions which will allow most programs to refer to these locations for common work space. The 6800 instruction set was copied as closely as possible from the PDP-11 with a shorter word length.

d. Software Support

Motorola is developing a cross-assembler, interactive simulator and a PL/M type high-level language.

e. System Considerations

The 6800 provides for maskable interrupts at four levels. All the I/O devices appear at one interrupt level so that once an interrupt is detected, polling must be used to isolate the device requiring attention. When an interrupt occurs, the processor places status and register contents into the push down stack automatically. The interrupt return instruction refreshes the state of the machine from the stack. A software equivalent of the interrupt is provided too; the computer's state is pushed onto the stack, and an

interrupt service routine is invoked. DMA is provided through a Three-State Control Signal. The 6800 is a microprogrammed machine but is not microprogrammable by the user.

f. Comments

The 6800 is a compromise of the powerful PDP-11 and an eight-bit machine. It has a good instruction set with excellent conditional branching. The major disadvantages are its slow speed and lack of software support.

3. General Electric CRD-8

a. Hardware Considerations

The CRD-8 is an eight-bit parallel processor unit currently being produced by General Electric on a two-sided printed circuit board approximately 4 x 8 inches. This card contains approximately 60 standard $T^2L$ devices including 1K of Program RAM. Typical power dissipation is 18 to 20 W. Figure 30 is a block diagram.

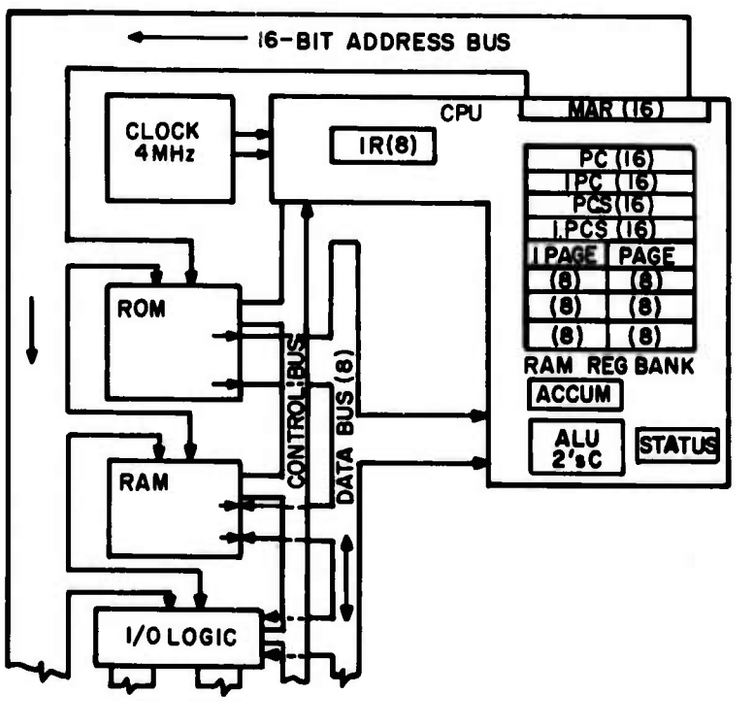



Figure 30. GE CRD-8 Block Diagram

The CRD-8 CPU operates as a signed 2's complement machine. Data access to the machine is page oriented. For many instruction types an eight-bit page register located in the RAM register bank provides the upper eight bits of the 16-bit address specification. In addition to the eight-bit page register, a 16-bit program counter (PC) and program counter save (PCS) are located in the RAM register bank. The program counter save contains the return address from subroutine calls. The PC, PCS and PAGE registers have duplicate counterparts for interrupt programs. The interrupt structure is such that when an interrupt is processed, the IPC, IPCS and IPAGE registers are accessed instead of the PC, PCS and PAGE. Return from interrupt reverses this register swap. The remainder of the RAM REG Bank contains three 16-bit registers.

The CRD-8 operates at a clock frequency of 4 MHz. Typical instruction time is approximately 2 $\mu$s, based on a memory access time of 100 ns.

b. Hardware Support

GE offers a standard I/O board which provides proper I/O multiplexing on the data bus. Decoding logic for Teletype I/O is included. A standard maintenance panel is also available providing an operational processor unit that contains the CPU board, I/O board, and memory cards. A power supply package for a 1K-8K RAM processor is also available.

c. Software

The data and instruction words are eight bits in length, while the address bus is 16 bits wide. The lower half of the three RAM BANK registers can be used in arithmetic operations while the full 16 bits of each register can be used as memory addresses. This provides indirect addressing with the added options of register post increment (after indirect usage) or register predecrement (decrement register prior to indirect usage). All incrementing and decrementing is handled in the eight-bit ALU which causes "roll-over" at all page boundaries in those applications where 16-bit addresses are incremented beyond a page boundary. The instruction repertoire contains extensive branching capabilities on the various condition flags (overflow, carry, zero). Expanded I/O capability is available as a hardware option (Phantom I/O) providing a full page (256) of I/O ports which are accessed via memory read/write instructions.

d. Software Support

GE has developed a full set of software support for the CRD-8 including an Assembler, Compiler (8-BOL, similar in structure to PL/M),

Macro Simulator, Microcode Simulator, and CRD-8 Software for program loading and debug.

e. System Consideration

Extended temperature range operation as well as other improved environmental specifications are possible by utilizing 5400 series $T^2L$ in place of the commercial grade components. New instructions can be implemented easily by reprogramming the control ROMS. This is not possible with other chip type processors which have the control ROMS on chip.

f. Comments

The CRD-8 processor is unique in that it is a processor design implemented in standard $T^2L$ rather than a special LSI chip, with associated peripheral logic. There are distinct advantages and disadvantages to such an approach:

Advantages:

(1) Modifications to the CPU architecture can easily be implemented.

(2) A Mil-Spec processor capability with a minimum of redesign.

Disadvantages:

(1) Higher cost and power consumption than an LSI equivalent.

(2) Larger in size than CPU chip-type processors.

General Electric has an agreement with Solid State Scientific in which $S^3$ will implement the GE design using CMOS/SOS. Samples should be available in late 1975.

4. Intersil IM 6100

a. Hardware Considerations

The IM 6100 is the first microprocessor incorporating a 12-bit CPU on one chip. The CMOS chip requires a +3 to +7V supply voltage. All signals are TTL compatible. Intersil boasts a low power of 10 mW for typical

operations. All that is needed for a clock is a crystal (rest of needed clock circuitry is internal to the chip). Figure 31 is a block diagram.

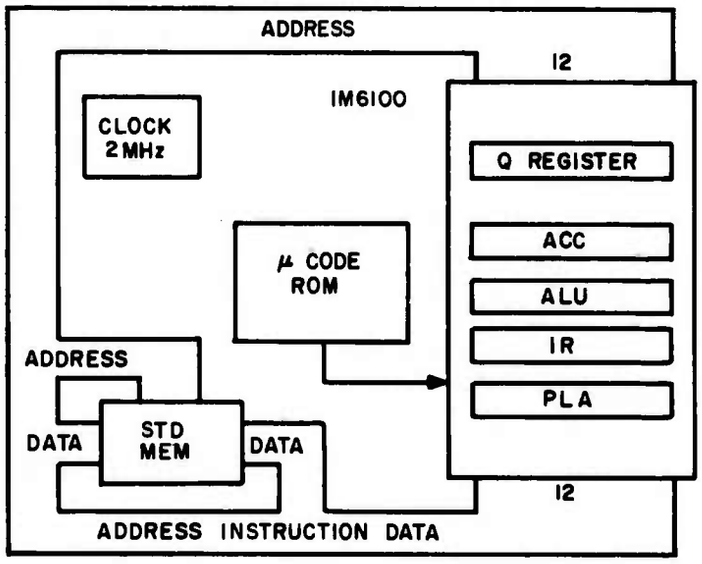



Figure 31. Intersil IM 6100 Block Diagram

The IM 6100 architecture emulates the Digital Equipment Corporation PDP-8. The CPU has six 12-bit registers: Accumulator, Multiplier-Quotient, Program Counter, Memory Address, Instruction and a Temporary Register. Operations are decoded in a PLA. The CPU has direct, indirect and autoindexed memory addressing. The IM 6100 recognizes all of the PDP-8 software. It can handle 64 I/O devices and 4K of memory directly (32K with extended memory control hardware). It has DMA capability and can handle any type or speed semiconductor memory in any combination. In short, the IM 6100 is a PDP-8 miniaturized in a 40-pin package. A 12-bit memory to accumulator ADD is accomplished in 5 $\mu$s.

b. Hardware Support

No hardware support is supplied by Intersil with the exception of semiconductor memories. However, any support DEC produces for the PDP-8 is compatible with the IM 6100.

c. Software

The IM 6100 has over 40 instructions. It recognizes the instruction set of the PDP-8 including AND, TAD, ISZ, DCA, JMS, JMP, JOT and OPR. One notable omission is a subtract operation. This must be done using 2's complement and adding. Conditional branching can only be done on zero accumulator, non-zero accumulator, negative accumulator and positive accumulator values.

d. Software Support

Since the IM 6100 is software compatible with the PDP-8, any support software the PDP-8 has can be used for the IM 6100. The PDP-8 is a mature machine and has a large amount of support software available. There is some type of legal agreement for software between Intersil and Digital Equipment Corporation.

e. System Considerations

The IM 6100 and the PDP-8 are software compatible. The differences in the hardware interfaces are described below:

(1) IM 6100 handles up to 4K 12-bit words of memory.

(2) The Extended Arithmetic Element option of the PDP-8 cannot be used with the IM 6100.

(3) DMA structures are different.

(4) The IM 6100 control panel is treated as a programmed I/O.

(5) The IM 6100 may have a completely "transparent" bootstrap loader.

f. Comments

The IM 6100 is a PDP-8 with the above exceptions. One problem with the IM 6100 is that in the PDP-8, a subroutine call modifies the first word of the subroutine; the return address is stored there. This will not work if subroutines are stored in ROM, as in many microprocessor applications. The lack of a subtract instruction is a minor difficulty which can be solved with more memory and slower execution.

5. Intel 3000

a. Hardware Considerations

The 3000 family is a microprogrammable processor implemented in two-bit slices. The 3002 is a two-bit slice while the 3001 is a microprogram control unit. The two devices require a single +5 V supply with 0.750 W for the 3002 and 0.900 W for the 3001. Typical room temperature speeds of 160 ns per microinstruction can be achieved. Figure 32 is a block diagram.

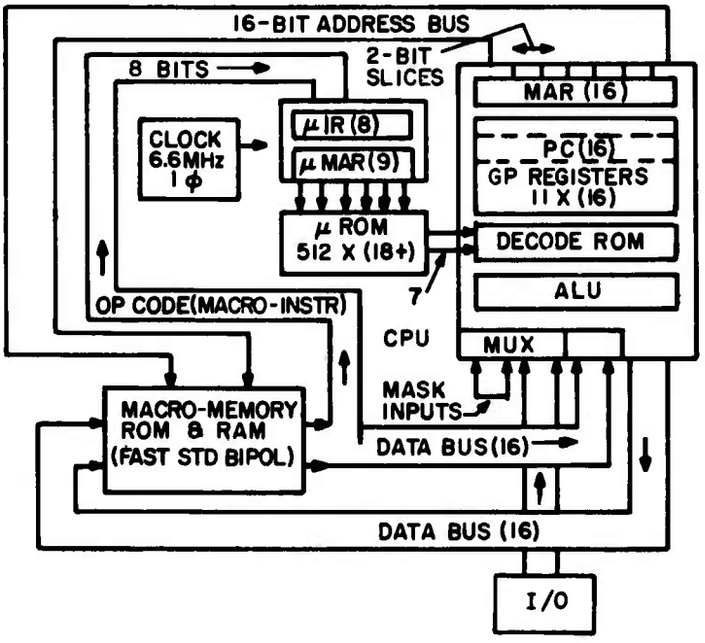



Figure 32. Intel 3000 Block Diagram

The architecture and microinstruction set have provisions for adding additional hardware to expand its capabilities. The microprogram is stored in conventional bipolar ROM's and pipeline registers can be added to increase speed. The 40 pin 3001 generates the ROM address sequences by analyzing the current microinstruction, status flags and up to eight bits of the macro instruction fetched from main storage. Extensive control logic includes four-way and sixteen-way jumps that are steered by the microinstruction bits. The MCU stores the ALU flag bits and selectively re-inserts them on succeeding cycles to implement some functions.

The MCU has no provisions for saving or restoring the current microinstruction address. This weakness eliminates microlevel subroutines which are a serious inconvenience. Attempts to save the address externally will be hampered by the fact that only eight of the nine address bits can be loaded back into the chip. There are ways to implement microsubroutines with external hardware but they all appear to be difficult.

The 28-pin, two-bit slice allows both ripple and look-ahead carry; however, a special look-ahead carry generator is required. The 3002 has only a carry out flag. Sign and zero flags can be generated with external hardware in the same microcycle. The overflow flag cannot be generated within the same microcycle. This is a serious problem in implementing a minicomputer. The 3002 has 10 general registers (no read/write capability in the same cycle) but does not have a Q register. The 3002 has a K bus which supplies bit masking capability.

b. Hardware Support

Intel provides a 16-bit minicomputer kit for $750 which provides most of the essential components for the computer. They also have a line of computer oriented components (priority interrupt control, registers, etc.) which will supplement their computer products. Intel has no plans to introduce a general purpose macro instruction set (like 8080) and thus has no plans for a front panel, rack, boards, etc.

c. Software

The instruction set is developed by the user for a given architecture. The slice responds to over 40 useful functions - arithmetic, logic, shifting, incrementing, decrementing, bit/byte manipulators are achieved on the 3002.

d. Software Support

Intel provides a symbolic microassembler for development use.

e. System Considerations

Interrupts, I/O, Addressing, DMA, etc., would have to be developed by the user.

f. Comments

The inconvenience of the slice is that it is only two bits (more chips required per word length than a four-bit slice) and it does not generate

all flags needed for minicomputer operation. Not having a Q register or read/write capability into the general registers in the same microcycle results in a slow multiply/divide operation. The 3001 MCU also has problems in generating addresses (diagonal transportation in both memory page and location is impossible in one microcycle). Another disadvantage is the microlevel subroutine weakness. These disadvantages add up and produce a relatively slow (compared to microinstruction speed capability) minicomputer. However, if the goal is not a minicomputer but a data controller, the bit/byte manipulations produce a very desirable machine.

6. Monolithic Memories 5701

a. Hardware Considerations

The MMI 5701 is a Register/ALU four-bit slice. It contains no circuitry required for instruction fetching and decoding. The slice contains an arithmetic logic unit (like the 74181), a 16 register file, shifting capabilities and a Q register used to extend the accumulator to double precision. There are two four-bit address lines to select the register file. The register file has separate read/write capabilities such that both functions can be achieved in one microcycle. The maximum cycle time over the full temperature range is 250 ns. One +5 V supply is needed with a single-phase clock. Maximum power required is 1.25 W. Figure 33 is a block diagram.

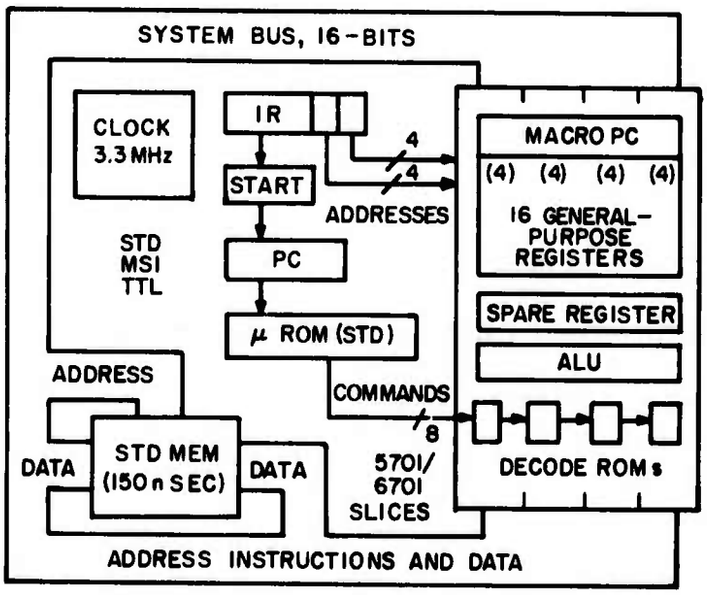



Figure 33. MMI 5701 Block Diagram

Control of data flow requires 16 bits: two four-bit addresses for the register file and eight instruction bits which are decoded into 17 control signals on the chip. There are also four-bit input and output ports for communicating with a bus system. The slices are designed to be cascaded to derive machines of any word length. Carry propagate and generate can be used with look-ahead carry generators to improve the speed performance of larger systems.

The four-bit slice produces four flags each cycle. They are overflow, sign, zero detect and carry. These flags are essential for use in a minicomputer system. The Q register allows easy implementation of multiply/divide algorithms. Thus, due to the Q register, flags and dual read/write register modes, the MMI 5701 allows a minicomputer architecture to be implemented in fewer cycle steps than other available slices.

b. Hardware Support

Monolithic Memories offers an MMI 300 microcomputer that emulates Data General's Nova 1200 series line. A wide range of boards, chassis and debug packages are available from MMI.

c. Software

The instruction set is developed by the user for a given architecture. The slice responds to 32 microinstructions. Arithmetic, logic and shifting is easily achieved; however, bit/byte testing and manipulation would be achieved externally to the chip.

d. Software Support

Debugging software is available for the MMI 300; however, actual programming support is nonexistent.

e. System Considerations

Interrupts, I/O, Addressing, DMA, etc., would have to be developed in the architecture by the user. This gives the user the flexibility to design a minicomputer which is tailor-made for a specific application.

f. Comments

The 5701 is one of the few military microprocessor devices available. What it lacks in speed, it compensates for in architecture. It readily yields a minicomputer architecture. Its only major disadvantage is lack of a control circuit design and the difficulty in manipulating data bits and bytes.

7. Texas Instruments SBP0400

a. Hardware Considerations

The SBP0400 is a four-bit RALU slice. It is the only microprocessor product that uses $I^2L$ technology. $I^2L$ yields the best density of the known technologies. T.I. has boasted execution times of 350 to 530 ns,, however, these speeds have not been achieved as yet. The power supply is not a voltage but a current supply. This is achieved by a voltage supply and a current limiting resistor in series with the slice. The slice requires a maximum of 250 mW of power. All inputs and outputs to the 40-pin package are TTL compatible. A single-phase, 2 MHz clock is required. Figure 34 is a block diagram of the SBP0400.

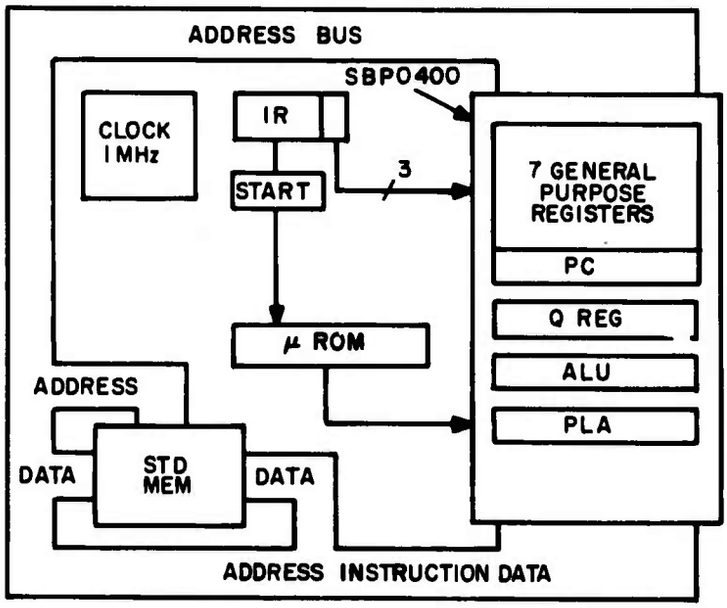



Figure 34. SBP0400 Block Diagram

The slice has seven working registers and a program counter. The accumulator has a Q register for double precision arithmetic. Shifting can be accomplished before the A or Q register via a multiplexer. The nine-bit operation word is decoded in the slice by a PLA. The 120 bit-output of this PLA may be latched with an external clock. This allows the external circuitry to address a new instruction while the slice performs the present operation. This technique is called PIPELINING and is used to achieve a high

speed capability. The program counter may be incremented to the next instruction after the PLA is latched. The chip has a problem in the flags that are present. No overflow flag is provided and the sign flag must be generated externally.

b. Hardware Support

None.

c. Software

The instruction set is developed by the user for a given architecture. The slice responds to 459 nonredundant operations. Arithmetic, logic and shifting are easily achieved. Multiplication can be accomplished by using the accumulator extension register.

d. Software Support

None.

e. System Considerations

Interrupts, I/O, Addressing, DMA, etc., would have to be developed by the user. This gives the user the flexibility to design a minicomputer tailor-made for a specific application.

f. Comments

Even though $I^2L$ is not as fast as $T^2L$, speed can be improved by using the pipeline register scheme internal to the chip. Chips can be cascaded using carry look-ahead generators. The Q register and $T^2L$ compatibility with $I^2L$'s low power make the chip very useful. The lack of flags and slower than $T^2L$ speeds are the only disadvantages.

## B. MICROCOMPUTER FORECASTS

Extrapolated from current technology, predictions for 1980 would indicate considerable improvements in LSI capabilities as shown in Table 20.

Assuming the projections are correct, for a few hundred dollars, a minicomputer of the DEC PDP-11 class could be configured with the CPU on a single chip. Similarly, packages containing either 4K x 16 bit RAM or ROM, an I/O controller or Peripheral Interface Adapter (PIA) can be envisioned.

TABLE 20. LSI CAPABILITIES

| LSI Parameter | 1966 LSI | 1973/1974 LSI | 1980 (projected VSLI) |
|---|---|---|---|
| Performance | | | |
| Clock rate (max) (MHz) | 5 | 30 | 100 |
| Speed-power product (pJ) | 100 | 3 to 10 | 0.1 to 1 |
| Complexity | | | |
| Chip size (max) (mil) | 100 | 250 | 500 |
| Device density (devices/chip) | 50 | 5000 | 200,000 |
| Functions | | | |
| Random-access memory | 16 bits | 2048 bits | 64,000 bits |
| Random logic | 4 gates | 500 gates | 10,000 gates |
| Minicomputer CPU (bipolar) | - | 10 chips at 5 MHz | 1 chip at 50 MHz |

To illustrate the effect of these projections, consider the present day LSI-11 microcomputer, which can emulate the PDP-11/40 minicomputer. The CPU is contained in four NMOS LSI devices, housed in 40-pin packages. The 8.5 x 10 inch printed circuit board also contains a 4K x 16 RAM, a 16-bit I/O port, as well as interrupt, clock, and power-on reset logic. With 1980 technology, the present 60 DIP, 85 square inch design could be reduced to 4 DIP's and 15 square inches.

Looking deeper into the crystal ball to the mid-1980's, one might expect to have 3/4 inch square LSI chip with up to $1/2 \times 10^6$ active devices. This would make it possible to fabricate the CPU, a 4K x 16 electrically programmable non-volatile memory, a universal interface adapter, DMA control, bus control, a clock and timing circuits onto a single chip. This chip could be a truly universal microcomputer similar to present-day minicomputers, having high speed, ultra-reliability, low power dissipation, low cost and small size, all contained in a 1 inch x 2 inch, 40-pin DIP package. This universal chip would provide preprocessing, central processing and post-processing computer capabilities. A combination of these could be structured as a distributed computing architecture that would satisfy most medium performance aircraft and spacecraft computing requirements.

## C. MICROCOMPUTER ARCHITECTURE CONSIDERATIONS

### 1. Objective

The design and implementation of a microcomputer system which must perform satisfactorily in a hostile radiation environment requires a set of hardened functional elements. Besides this hardening constraint, it is desirable to design a computer architecture which allows varying computational performance. Flexible performance provides a means of standardizing hardware elements on cost effective LSI designs while allowing a reasonable match to a spectrum of military applications.

The bandwidth of computing applications for space and aircraft missions in many ways presents a dilemma. Present LSI microcomputer technology provides limited computing/performance capabilities which demands a compromise of some of our objectives.

### 2. Limitations

A microcomputer is defined as a stored program computer system in which the central processing unit (CPU) consists of a small number of LSI circuits. The only requirement of this definition is the exclusive use of LSI in the CPU. This definition is not based upon performance or word size. A microcomputer should not be confused with a microprocessor. The word microprocessor has at least two commonly accepted meanings. It can either denote a microcomputer or a computer that is microprogrammed. When used in the latter context, some, but not all microcomputers are microprocessors.

Some microcomputer systems that are readily available today are based on some form of MOS LSI technology and are designed for nonmilitary uses. The associated logic speed is modest compared to the bipolar Transistor TransistorLogic (TTL) devices commonly used in minicomputers and large scale computers. In practice, microcomputer designers are using logic speeds that are low compared to those used in other products. Emphasis has been placed on rather conservative technology. Functionally and computationally, the application user derives modest performance capabilities.

The economics of circuit production place a limit on the maximum silicon chip area. Microcomputer designs suffer from this constraint in at least two ways. First, much of the logic of a microcomputer is not the highly regular form that yields high chip density. Therefore, the total number of logic functions available is limited. Second, the chip area required to buffer signals off-chip is several times greater than that required on-chip. Because of this, designs that minimize off-chip connections are favored. The net result

of these limitations is that designers are unable to include all of the features and power normally available in conventional computers. Many more or less standard performance enhancement features, such as complete interrupt systems, index registers, address modifications and indirect addressing tend to be excluded. Of course, the equivalent of these features can be performed at the expense of extra execution time. Current microcomputer designs provide modest speed and throughput capability. Typically, cycle times are in microseconds with relatively fast register-to-register operations, and slower, multiple cycle operations when memory accessing is involved. The performance of today's microcomputers is roughly comparable with the minicomputers that were available more than 10 years ago.

3. Application Impact

The basic task of all computing machines is the efficient execution (in time and cost) of the data management and computing tasks of the intended application. It is not surprising to find order of magnitude speed differentials between one machine and another. Specialized designs for an application tend to be superior to the general purpose machine. In general, a design choice is based upon cost, flexibility, reliability, and growth potential for new applications. The general purpose machine generally is the user's choice in such tradeoff comparisons because of these factors.

The selection of a candidate computer system could start with a principle called an architectural structure matching. The match consists of considering elements in design and combining them in a way to complement one another so that an optimal cost effective computer design may be achieved.

Five primary structural elements are subject to tradeoff criteria. These are: 1) technology, 2) algorithm, 3) data, 4) programming language and 5) architectural units. These are discussed in the following paragraphs.

a. Technology

Technology tends to dominate computer machine design thinking. The basic speed and functional blocks available inevitably condition the feasibility of the design. Some of the aspects of technology, such as logic family performance and hardness were discussed in previous sections. Other aspects are the system interconnection circuitry, level of modularity, the availability of multiple functional modules and specialized arithmetic hardware to supplement primitive computing functional elements. These factors enable the utilization and specialization of dedicated computer operations. In the next section, a few applications are discussed and the effects of special computing features which are important to their effective execution are examined.

b. Algorithms

A variety of digital computer computational algorithms exist and are applicable to a wide class of military oriented problems. Three data processing tasks were discussed in Section II. Kalman Filtering, Radar Beam Steering and GPS Vehicle Navigation were selected as a set of typical space and aircraft computing operations. It was implicitly assumed that they could be performed in a dedicated manner. The three tasks were partitioned from a general class of command and control computing problems. Examination of the computational operations associated with the three functions indicate the need for some common instruction capabilities. A reasonably sophisticated multiply-add and trigonometric function generation facility is required for all three. All three applications would benefit from fast, efficient vector, matrix and data manipulation instruction repertoires. Two of the three require co-ordinate transformation operations requiring possible matrix inversions, a basic multiply, divide and add computation. Some applications allow for polynomial multiply operations and look up tables to substitute for divisions in co-ordinate transformations. Good address preparation and modification facilities dominate in this area.

The Beam Steering problem emphasizes the need for high speed data movement from vehicle interfaces and high speed, minimum instruction set requirements. The Kalman Filter and the GPS navigation problem do not require much data movement but need high precision "number crunching" with medium performance compared to the Beam Steering problem. Optimizing each application's needs and having them complement one another leads to the acceptance of current minicomputer design concepts. Some compromise in this area is necessary and one approach is to tailor each computer's hardware with standard building blocks. To be more generic and to encompass more applications, some further discussion of digital filter algorithm is beneficial.

Recursive filters implemented by digital computers are receiving more acceptance with the improvement in computer performance and reduction in costs. Any order filter can be implemented in hardware as an interconnection of second order sections which are mathematically represented by the function:

$$Y_n = AY_{n-1} + BY_{n-2} + CX_{n-2} + DX_{n-1} + X_n$$

This function represents the basic operations of a whole set of basic transforms, beside having a counterpart in Kalman filtering and polynomial expansions and approximations. If the coefficients and variables are considered vectors, we have the core computation which the dedicated efficient microcomputer executes.

Examination of the Euler transformations associated with vehicle navigation indicates the usefulness of a high speed, simple method to generate trigonometric functions. The Cordic algorithm in which shifting and adding is used to generate most of these functions rapidly and simplistically can be developed in a dedicated microcomputer and pipelined into the major tasks of filtering when necessary. Where time is not critical, trigonometric functions can be generated by the standard power series expansions for the desired range of precision. The iteration intervals to complete required algorithms have been given for the three tasks. From this information one can appreciate that one more dedicated microcomputer is needed. The Beam Steering problem requires the largest number of simple computations. Although limited arithmetic precision is required, the speed of the operations saturates any contemporary microcomputer.

c. Data Structure

Some of the impact of data structures has been implied in the discussion of algorithms. For the second order filter sections, the number of bits in the data word comprising the coefficients is directly related to the variable word size. The number of multiplications and adds per function and per unit time affects the precision obtained in arithmetic processing. A variable word size matched to the application is most cost effective for the application but not necessarily for the hardware. Most problems could be solved with a 16 bit double precision arithmetic capability or 24 bit single precision capability. Eight bit machines generally pay the penalty of excessive execution time and may not even be able to accomplish the required tasks. Processor word length is clearly related to the application area, e.g., 12 bits generally for radar, 16 to 18 bits for communications and 16 to 32 for navigation and guidance. These choices affect scaling, speed, frequency of overflow, memory size, multiplier time, truncating, rounding and overall system cost. Using a microcomputer to cover such a range of requirements requires considerable compromise.

Memory structure and memory data access are also areas for tradeoffs. Partitioned memories, separate instruction and data caches between program and arithmetic units and scratch pad banks acting as part of the Arithmetic Logic Units have their roles in improving computer performance. Expandable microcomputer system designs evolving out of current minicomputers are trying to benefit from past experience by incorporating these features.

Floating point is a desirable add-on feature for programmers and may be required for the sophisticated navigation problem, which requires large numbers of vector and scaling operations.

d. Programming Language

Examination of the current and projected microcomputers indicates a minimum, in many cases, and complete lack, in other cases, of software support. For the stand alone microcomputer system, reasonable software support is available if it is an archetype of a vendor's previously conceived computer family. Building your own architecture and developing your own instruction set through microprogramming at the most primitive level is a tedious time-consuming task. Most microcomputer programming is constrained to the assembly language level; however, only a few vendors at present are providing this capability. For the present, higher level language programming of microcomputers does not appear to be efficient enough to warrant consideration. One exception is the 8 BOL language developed by GE for the CRD8 microcomputer. Its uniqueness and efficiency is predicated upon its very close match to the hardware implementation. This probably will be the trend for microcomputer systems. Another approach would be to establish a set of primitives for special purpose processing algorithms when the microcomputer is in a very dedicated computing environment.

e. Hardware Architecture

Four factors discussed previously are useful in appraising a computer vendor's product. From a generic problem perspective based on the candidate applications, the system structure should be modular in nature and should be built from a set of microcomputers either in a single or multiple chip microprogrammed form. Each of the microcomputers should have a flexible, variable data and word length of one to four eight-bit bytes, with flexible direct and indirect memory addressing and moderate performance arithmetic capability. The modular elements that comprise memory and input-output are assumed to be compatible with the central processor and peripherals.

With these characteristics as a baseline and with a suitable microcomputer, a higher level computing organization may be considered (using a systems bottom-up approach versus a system top-down approach which also is feasible). Tradeoffs to be made in the specification of the hardware include specialized memories for data, microprogram, and macro program storage, specialized arithmetic units, such as blister multiply or floating point units, hardware parallelism and instruction pipelining, and the speed, power and cost tradeoffs between sequential and combinational logic implementations.

## 4. Benchmark Characterization

A tradeoff analysis to determine which microcomputer to use in a given application would be difficult to conclude, due to the proliferation of devices with various architectures and attributes. A typical approach to narrow the choice is to apply benchmark programs to the candidates to determine processing time and memory requirements. Typical benchmarks would measure the time and memory required for movement and manipulation of data, storing and retrieving status upon interrupt, searching for parameters in memory, monitoring peripheral devices, number crunching, etc. As one could guess, the closer the microcomputer approaches a sophisticated minicomputer in performance, the more likely choice for the applications.

One criterion in the selection process is instruction power. As an illustration, consider the effects of adding two eight-bit quantities stored in memory and returning the sum to memory. The sequence of instructions to perform this operation takes 177.5 $\mu$s using the Intel 8008. Now suppose this instruction is to be stored in a list. In a conventional machine which has an index register, a single instruction would accomplish this. The sequence takes 120 $\mu$s for the 8008. As a third example, suppose the contents of the accumulator are to be stored in an address specified by the contents of a pointer. A conventional machine would execute a single instruction in one memory cycle. The Intel 8008 could do this operation in 122.5 $\mu$s. A considerable time loss with the 8008 is obvious compared to a conventional machine.

Instruction power improvement is obviously obtained when the Intel 8080 is considered. This machine is similar to the 8008 but there are several improvements. Perhaps the most important is the execution time. The improvement varies with instructions, but the 8080 is on the order of six times faster than the fastest 8008. The most obvious change is the bus structure. The 8080 has a separate 16-bit memory address bus, which avoids the time consuming need for multiplexing a single bus. The separate bus also eliminates the need to latch the address and the associated logic. Through improvements in addressing, stack point and word length movements, considerable execution efficiencies are achieved. Trial programming gives execution times for the add to memory problem of 24 $\mu$s, the store in a list takes 17 $\mu$s and the store at a pointer takes 21.5 $\mu$s.

Both the 8008 and the 8080 are deficient in their ability to perform multiplication. An eight-by-eight bit multiplication giving a 16-bit product using the 8080 takes about 260 $\mu$s. Since this time is not acceptable for most number crunching applications, one solution is to provide specialized multiply hardware blisters. The time can then be reduced to approximately 20 $\mu$s using efficient instruction coding.

Benchmarking a set of computers for an application can be a tedious job. A thorough, working knowledge of the instruction set of each candidate is required. A subject of great controversy, benchmarking has been called a black art in decision making by computer experts.

## D. ARCHITECTURE SPECIFICATION

### 1. General Requirements

Considering the application analyses conducted during this study and anticipating other present and future applications, certain candidate computer selection criteria are evident. To achieve cost effective utilization, a hardened microcomputer system should meet the following constraints:

- operates in a military environment
- readily available
- low system cost
- highly flexible
- high speed
- low power
- support software available
- instruction sophistication
- radiation resistant

Rather than designing new hardware and support software, the approach of using an existing design with documented software is recommended. The user can realize lower costs and quicker deliveries with this latter approach. Although hardening constraints might require modification of the chip layout or electrical design, retaining the overall hardware architecture will still provide considerable benefits to the user.

The available devices can be classified as either byte slice or fixed word length machines. Sixteen bits is the current maximum for the fixed word length devices. The applications which were examined require numerical computations with accuracies of 16 to 32 bits. In order to achieve the longer word lengths with the current technology, a byte slice architecture is indicated. The smaller, four-bit and eight-bit, fixed word length machines would be more

suitable in other applications, such as dedicated controllers, command sequences, and slow speed interface units. These devices could have their place in hardened systems. The bit slice machine is not the best answer for all applications, but it provides the necessary flexibility to the designer to satisfy longer word length requirements.

For the real time, numerical computation applications, execution times on the order of one microsecond are required. This figure lies between the extremes of that required for real time signal processing (approximately 100 ns or less), and for controllers (approximately 10 $\mu$s or more).

The bit slice architectures do not include instruction decoding logic on the chip in the same manner as the fixed word length devices. An external control chip, Control Read Only Memory (CROM), is required which may be microprogrammed by the user. In this manner, the user can specify his own macro instruction set, tailored to his application with the appropriate level of sophistication.

The level of radiation hardness depends on the application. The requirements for various applications are shown in Table 21. The ability of the indicated technology to satisfy the requirements is shown in Table 22. A dielectrically isolated, low power Schottky TTL technology would meet the radiation resistant, low power and high speed criteria stated initially.

TABLE 21. RADIATION HARDNESS REQUIREMENTS

| Environment | Application 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Neutron Fluence (1 Mev Equivalent) | $10^{10}$ | $10^{12}$ | $10^{12}$ | $3 \times 10^{14}$ |
| Work through Dose Rate (Rad-Si/s) | - | - | $10^{8}$ | - |
| Survival Dose Rate (Rad-Si/s) | $5 \times 10^{6}$ | $10^{8}$ | $4 \times 10^{11}$ | $5 \times 10^{12}$ |
| Total Dose (Rad-Si) | - | $5 \times 10^{3}$ | $10^{6}$ | $10^{6}$ |

TABLE 22. TECHNOLOGY HARDNESS CAPABILITY

| Technology | Application 1 | 2 | 3 | 4 | Limit |
|---|---|---|---|---|---|
| CMOS | X | X | | | $\dot{\gamma}_s = 10^{10}$ |
| CMOS/SOS | X | X | | | $\gamma = 3 \times 10^5$ |
| TTL | X | X | | | $\dot{\gamma}_s = 10^{10}$ |
| DI/TTL | X | X | X | X | |
| IIL | X | X | | | $\dot{\gamma}_s = 10^{11}$ |
| ECL | X | X | | | $N = 10^{12}$, $\dot{\gamma}_s = 10^{10}$ |
| EFL | X | X | ? | ? | $\dot{\gamma}_s = ?$ ($> 9 \times 10^{10}$) |

2. Architecture Recommendation

Characteristics of the various microprocessor products from over 31 vendors who manufacture over 48 different types have been examined. The products have been categorized as to their manufacturing technology, bit size, CPU/slice/hybrid availability, number of instructions, machine cycle time, I/O speed, memory capabilities and software support. This examination/evaluation was accomplished under a GE Standard Avionics Microprocessor Development Study initiated in 1974 and is still an on-going project. As a result of this effort and the objectives expected to be obtained by utilization of hardened microcomputers in the present study, the following is recommended.

A distributed microcomputer system composed of byte slice structured microcomputers, using standard, basic building blocks is applicable to most of the tasks at hand and those visualized for the future. This will allow off-loading of highly sophisticated large scale data processing computers used in centralized data management organizations. Throughput will be enhanced, and fault tolerant characteristics will be achieved. Sparing of facilities using redundant, low cost LSI modules thereby becomes feasible. In those cases where sophisticated centralization is not necessary, a family of these standardized microcomputers can be tailored into functional systems and subsystems. The byte slice architecture offers the designer flexibility in satisfying various word length requirements. Radiation hardening requirements should be easier to meet than the computer on a chip device. Since hardening will probably require more silicon area, the chip manufacturer

can retain the same microcomputer architecture by going to a smaller slice in order to retain finite yields.

3. Byte Slice Comparison

Three candidate microcomputer byte slice architectures are under consideration since they represent a reasonable compromise of the desired characteristics. The choice of one over the others rests on the ability to develop a powerful instruction set using microprogramming hardware and interconnecting the basic building block computing elements effectively. A Data General Nova Class microcomputer using less than 50 LSI chips, including 4K/16-bit memory is feasible with MMI 5701 or the Intel 3002. Various characteristics of the slice devices are compared in Table 23.

The Intel two-bit bipolar ALU will not make a fast minicomputer. It cannot operate on two arbitrary registers at one time, and just about all minicomputers need this capability to be as fast as existing designs using Schottky ALU's and register files. Two register operation is also required for the memory addressing modes in most minicomputers.

The Intel 3002 will not be speed effective with existing 1 $\mu$s minicomputers and hence, will not efficiently emulate. The 5701 will be at least twice as fast as Intel at the macro instruction execution level, with four bits per unit versus two for Intel. (Typical macro instructions would be Register-to-Register Operation with indexed or relative addressing.) MMI power dissipation is 33% less than Intel, and MMI package area for four bits is 28% smaller.

Intel's microinstruction cycle is twice as fast as MMI, but MMI typically does two to four Intel operations in one cycle. Since each instruction must come from an off chip ROM, MMI will require at least 50% less ROM/PROM words than Intel for any minicomputer design. Intel's device is not a computer ALU like the 5701, but a controller building block.

Controllers, such as those for tape and disc units, do not require many minicomputer level instructions. They do, however, require bit testing, bit masking, and bit manipulations (such as set a bit, clear a bit, test a bit, etc.). Intel has a separate input port into their ALU which can permit bit manipulations. The 5701 can also do this, but the mask word must first be brought into the internal RAM, and then can be used in conjunction with the data in.

In systems requiring continual monitoring of various bits to indicate peripheral activity, Intel's unit will be faster than the 5701. Applications in the area include bus controllers, floppy disk and tape controllers, display controllers, etc.

TABLE 2. BIT SLICE COMPARISON

| Parameter | Intel 3002 | TI 0400 | MMI 5701 | |
|---|---|---|---|---|
| Slice Width | 2 | 4 | 4 | Bits |
| Power Dissipation (typical) | 750 | 112 | 1000 | mW |
| Internal Registers | 13-2 bit | 10-4 bit | 17-4 bit | |
| Logic Functions | | | | |
| AND, OR, NOT, EXCLUSIVE NOR | Yes | Yes | Yes | |
| TRANSFER | Yes | Yes | Yes | |
| FORCE 0000, 1111 | No | No | Yes | |
| Arithmetic | | | | |
| Increment, Decrement | Yes | Yes | Yes | |
| 2's Complement | Yes | Yes | Yes | |
| Add | Yes | Yes | Yes | |
| Subtract | No | Yes | Yes | |
| Shift Right | Yes | Yes | Yes | |
| Shift Left | Yes | Yes | Yes | |
| Zero Detect, Overflow Detect Flags | No | No | Yes | |
| Accumulator Extension | No | Yes | Yes | |
| Military Devices | No | Yes | Yes | |
| 16 x 16 Multiply (Add + Shift) | 16 | 20 | 5 | $\mu$s |
| ALU Cycle (worst case) | 100 | 1000 | 250 | ns |
| Register/Register | | | | |
| Add | 200 | 2000 | 250 | ns |
| Subtract | 300 | 2000 | 250 | ns |
| Exclusive OR | 200 | 2000 | 250 | ns |
| External ALU Input Ports | 3 | 1 | 1 | |
| External ALU Output Ports | 2 | 1 | 1 | |
| Effective Address Calculation | 600 | 3000 | 250 | ns |
| Typical 16 Bit Minispeed (no pipeline; load, store, register-to-register add) | 2.8 | 12 | 1.5 | $\mu$s |
| Application Area | Controllers | | Minicomputers | |

Intel also offers the 3001 microprogram control unit (MCU). This unit handles the sequence control section of the CPU. Intel's MCU is quite limited in capability since it is controller oriented. The unit cannot jump between arbitrary states, and cannot have microprogrammed subroutines. It does, however, save 20 TTL packages.

Intel's MCU is available, along with a priority interrupt chip (replaces five TTL packages), and a bi-directional bus driver (a few TTL packages). MMI's answer to the MCU is the PLA and FPLA. Intel's priority interrupt chip does not allow masking of interrupt levels or dynamically changing interrupt priorities, and is not well suited for interrupt structures in machines other than controllers.

Both the MMI and Intel devices use bipolar Schottky technology, while the TI device uses IIL. The speed-power tradeoff of IIL is apparent in the comparison with the Schottky devices.

Architectural deficiencies in the Intel device include the lack of a subtract instruction, no accumulator extension and the lack of external flags. The latter characteristic also applies to the TI device. These characteristics are important for fast execution, microprogram control, and efficient multiply/divide operations. Of the types considered in this study, the 5701 device has the best characteristics for the minicomputer application. The Intel device will provide superior performance in controller applications. The TI slice is best suited in low speed, low power applications.

## E. FAULT TOLERANCE

Providing the hardware and software necessary to achieve fault tolerant computer operations in a microcomputer system requires the utilization of many reliability enhancement techniques. Approaches which are applicable to fault tolerant design include intrinsic reliability of components, built-in-test-equipment (BITE), redundancy techniques, replacement and sparing, and fault masking. Any or all of these techniques may be augmented with software support. The techniques which are utilized in an application must be cost effective with respect to the criticality of the mission. Most vendors of computer products consider reliability enhancement of their designs as add-on features and do not incorporate fault tolerant characteristics at the lowest levels. The computer system architect attempts to achieve system fault tolerance at high modular levels. The higher the level in general, the more feasible the fault tolerant design and the more cost effective. With present technology, sparing

and replacement techniques are sufficient to give reliable fault free computer system performance over several years. A few contemporary LSI designs are purported to provide better than 100,000 hour mean-time-between-failures (MTBF). Several of these could achieve 95% reliability for as much as a five year period with adequate switching and replacement procedures. Duplicating and triplicating functions for error detection, isolation and correction must be justified on their own right as to their cost effectiveness in each application. Fault masking at the chip level is still at an experimental research level and requires considerable investigation before it can be suggested or utilized.

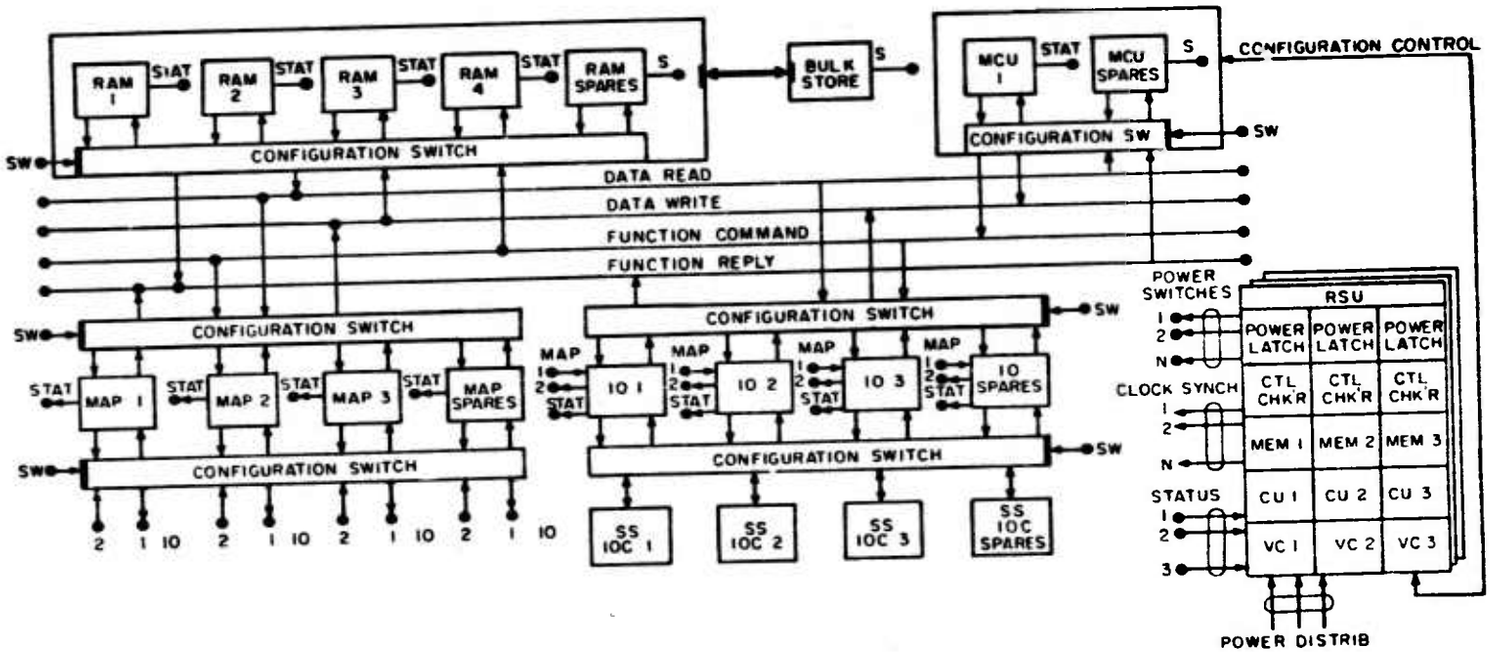
Up to this point, circumvention techniques, part of a fault tolerant hardened methodology, have not been discussed. Along with a selected architecture using a semiconductor technology, a magnetic memory technology may be necessary. Plated wire memory is available to provide fault tolerant backup. The safe-store of critical information may be essential to survive a temporary system power-down caused by incident radiation. A magnetic memory safeguard is necessary for critical programs, maintaining system status and for interim processing of critical tasks. A nonvolatile memory which stores bootstrap, and possibly roll back and recovery programs is essential to circumvention. Fault tolerant techniques must be designed into a system to achieve system recovery and survivability from transient failures and catastrophic stoppages. Means of detecting radiation effects in excess of intrinsic device tolerance are required to provide for timely enactment of circumventing procedures. Restart and recovery procedures which are part of the firmware and the nonvolatile fail safe stored software programs require tailor made instructions. The so called Program Triple Store instructions in which critical status data is stored in triplicate is a practical fault masking technique. Adequate hardware interrupt facilities provide a means of achieving prompt recoveries from failures. For imminent failures of a permanent nature, good hardware interrupt facilities provide a means of alerting the system promptly of the current failure so that a spare replacement procedure can be instituted. When spares are not available, some form of graceful degradation may be achieved by alerting another computer in the system, if one is available, to take on new tasks formerly assigned to the failed unit if that computer is not fully loaded; or if it is, to give up some nonessential tasks it is dedicated to, in order to permit it to do so. Graceful degradation techniques are highly dependent upon the multicomputing-multiprocessing computer organization configurations, and supervisory operating system structures. In the following sections a distributed federated organization of computers is discussed in which fault tolerance is achieved by graceful degradation along with sparing and replacement hardware strategies.

## F. MULTIPLE COMPUTER ORGANIZATIONS

Centralized and distributed multiprocessing computer organizations have been a subject of controversy and tradeoffs for many years. Each organization has its advantages and disadvantages. Until the evolution of the low cost LSI microcomputer, most computer users favored the centralized multiprocessing organizations because of proven efficiencies. Recent technological developments have caused the pendulum of favoritism to swing the other way. The low cost LSI microcomputer provides a means of developing a reasonable, cost effective, federated computer system. A group of federated microcomputers in which each has a dedicated task requires effective program and communication path partitioning between processing elements. Localization and dedication of specialized tasks requires a set of regional hardwired buses, each of which is assigned to the local microcomputer and its associated peripherals or subsystems. Control is assigned to a dedicated central computer, the function of which is to provide supervisory and data management control for the regional microcomputing activities. Although the federated system is recommended for a hardened microprocessor feasibility study and implementation, the centralized multiprocessor organization also can serve as a test bed.

Two diagrams of distributed processing organizations are presented for illustration using LSI microprocessing candidate multicomputers. Figure 35 follows a classical multiprocessor format in which the computing elements are independent and are configured to utilize common, shared resources. The multiprocessor has spares at the ALU, I/O and memory element level. This organization is best implemented centralized since all elements are functionally tied to one another. Throughput is variable since active processors depend upon computational load. When not serving as throughput enhancers, the modules becomes spares. The structure is noted for its graceful degradation and reliability enhancement features. A reconfiguration switching processor is replicated (TMR) and represents the hard supervisory core to provide switching and replacement strategies for the system to permit it to survive.

Figure 36 is an example of a set of stand alone, monolithic LSI microcomputers in a system in which federated supervision and task dedication is emphasized. The sparing and replacement level is a modular step above the previously discussed multiprocessing organization. Control of activities is assigned to a microcomputer, the counterparts of which are local microcomputers. Sparing and replacement of a failed module is a function of the hard core triple modular redundant (TMR) microprocessing unit. Bus structures for the centralized microprocessor organization are somewhat different than the federated system, in that the centralized system requires more sophisticated multiplexing and switching. Both approaches represent sophisticated system architectures and utilize LSI computer technologies.




LEGEND:
MAP = MAIN ARITHMETIC MODULE
MCU = MAIN CONTROL MODULE
IO = INPUT OUTPUT MODULE
RAM = RANDOM ACCESS MEM MODULE
RSU = RECONFIGURATION SWITCH MODULE

| CODE STATUS | |
|---|---|
| OPERATING | 10 |
| STANDBY | 01 |
| OFF | 00 |
| FAULT | 11 |

Figure 35. Distributed Multicomputer System - Candidate 1

Reliability enhancement techniques to achieve computer fault tolerance are illustrated by these organizations. In general, the architecture supports and supplements the microcomputing stand alone elements. These approaches are adjuncts to other techniques which include modular element reliability enhancement. A qualitative comparison of both organizations is provided in Table 24.

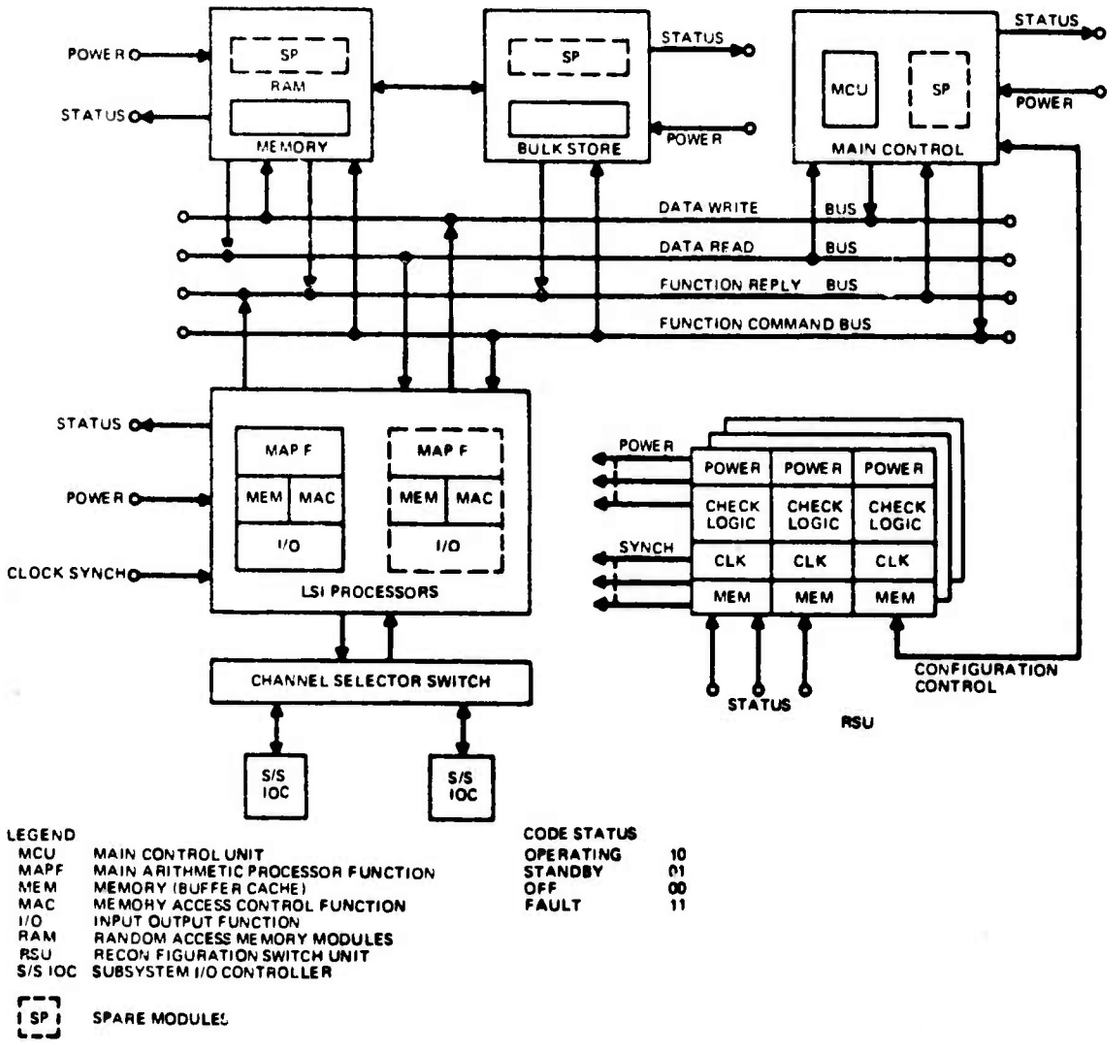



Figure 36. Distributed Multicomputer System - Candidate 2

TABLE 24. GENERAL ARGUMENTS RELATIVE TO MULTIPROCESSOR AND FEDERATED COMPUTER CONFIGURATIONS

| Characteristics | Multiprocessor Computer | | Federated Computer | |
|---|---|---|---|---|
| | For | Against | For | Against |
| **Hardware** | | | | |
| Reliability | Good for successive failures and degraded modes | Special vulnerability (in one location) | Low vulnerability (spacially dispersed) | For equal reliability requires more hardware |
| Maintainability | Excellent | | | May be difficult due to subsystem location |
| Efficiency | Low size weight, power, cost | Moderate memory interference | Low memory interference | Large size, weight, power and cost penalties |
| Flexibility and growth | Excellent | | | Poor |
| Integration/interfaces | Moderate - minimum interference | | | Difficult - many interfaces. Sequence dependent |
| **Software** | | | | |
| Executive program | Generalized and consolidated for centralized control and conflict resolution | More complexity | Less complexity | Specialized and dispersed, requiring a higher authority for conflict resolution |
| Applications programs | Easiest to control consistent standard interface | Less autonomy | More autonomy | Typically varying interface requirements |
| Flexibility-growth | Excellent | | | Very poor |
| Efficiency | Maximum - low overhead in memory and timing loads. Little duplication | Moderate memory | Low memory | Minimum - high overhead penalties in memory and timing loads. High duplication |
| Maintenance | Very good because of standardization | | | Typically poor because of minimum standardization |
| Simulation (system) | Straightforward - dependent upon executive and data bus | | | Complex - requires multiple executive asynchronous interaction |
| Emulation (software) | Moderate complexity | Can be quite large unless multiple emulation modes exist | Moderate complexity. Always constrained in size | Timing interactions from other computers must be generalized |
| Integration | Moderate complexity | Requires excellent software systems management | Low complexity. Requires average software systems management | |
| Languages | Compiler base language. Applications language based on compiler | Slightly greater inefficiencies in execution time and memory required | Applications dependent compilers. Slightly greater efficiency in execution time and memory requirements | |
| Degraded modes | Powerful - requires minimal intercommunication. Minimal time to change modes | | | Difficult - requires complex intercommunication. Moderate time to change modes |

## G. ARCHITECTURE EXAMPLES

### 1. High Speed ECL Processor

Although not available as a microprocessor in LSI form, the High Speed ECL Processor (HSP) is an example of the high speed which can be achieved from a simple architecture and ECL MSI parts. The speed of the processor makes it suitable for signal processing applications.

The block diagram of the HSP is shown in Figure 37, and its characteristics are summarized in Table 25. The hardware simplicity of the 16-bit processor is evident from the block diagram. Sophisticated features such as multiple general registers or a stack were not included to keep the design simple to allow economical use of processor modules in a multiprocessor organization. Basic instructions which do not require a memory reference are executed in 50 ns. The 2's complement multiplication of two 16-bit operands will yield a 32-bit product in 900 ns.

Motorola has announced a bit slice ECL processor which will be available in 1976. The family includes five devices: a four-bit processor slice, control, timing, memory interface and slice look ahead chips. The microprogrammable processor features a 75 ns cycle time over temperature and MECL 10,000 logic compatibility. When available, this processor will provide a significant addition to the high performance end of the microprocessor spectrum.

TABLE 25. HSP CHARACTERISTICS

| | |
|---|---|
| Word Length | 16 bits |
| Technology | ECL 10K MSI |
| Memory Capacity | 64K |
| IC's (w/o memory) | 92 |
| Power (w/o memory) | 32 W |
| Board Area | 100 sq. in. |
| Cycle Time | 50 ns |
| 16 x 16 Multiply | 900 ns |

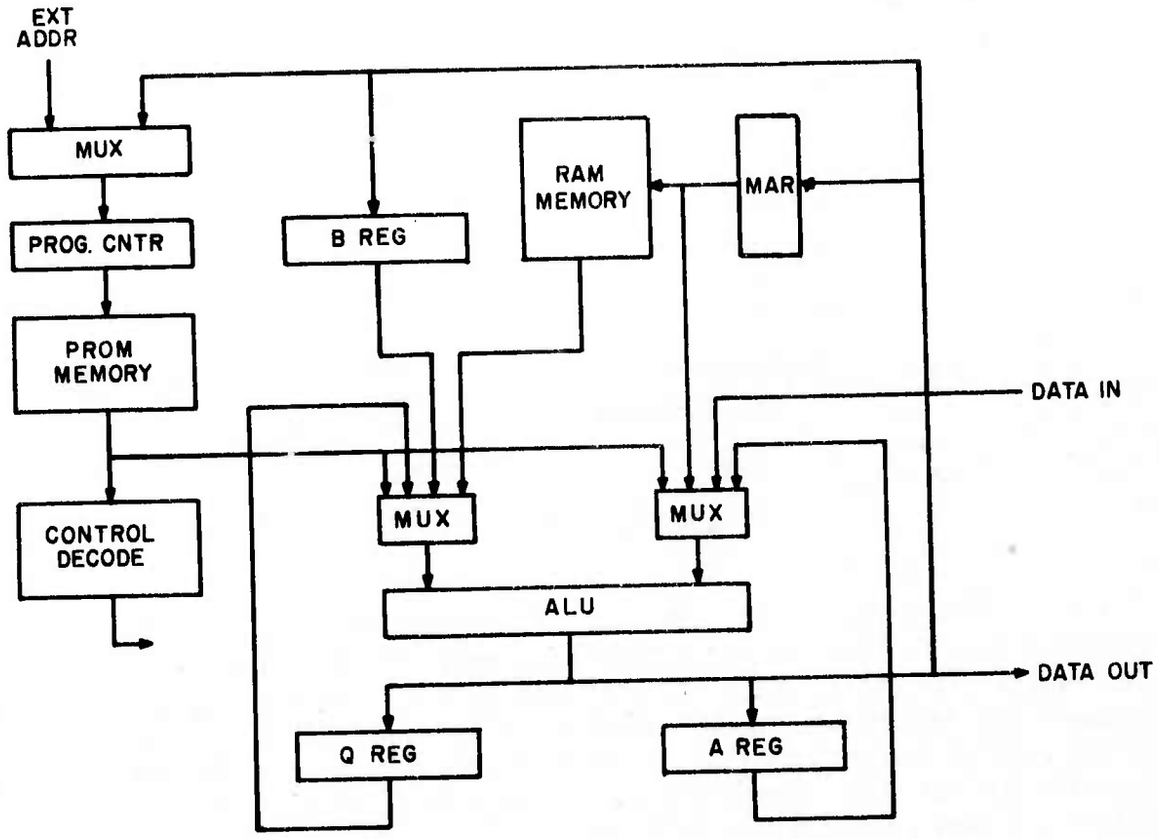



Figure 37. HSP Block Diagram

2. Bit Slice Processor

A block diagram of a processor which was designed for navigation, guidance and control computations is shown in Figure 38. Its performance is summarized in Table 26. The architecture for this machine was chosen to provide a basis for comparison to the bit slice machine (BSP) described in Figure 39 and Table 27. The navigation computer (NGC) contains plated wire memory for scratch pad storage, five general purpose and two index registers. The machine is microprogrammed and is several steps above the HSP in architectural complexity.

TABLE 26. NAVIGATION COMPUTER PERFORMANCE

| | |
|---|---|
| Word Length | 24 bits |
| Technology | TTL MSI |
| Volume | 236 cu. in. |
| Weight | 11 lbs |
| Power | 90 W |
| Instructions | 33 x, $\div$, sin, cos, $\tan^{-1}$ |
| Load/Store | 2.4 $\mu$s |
| Add | 3.6 $\mu$s |
| Multiply (24 bits) | 50 $\mu$s |

The general registers and arithmetic logic unit (ALU) are contained in the register and arithmetic logic unit (RALU) slices in the BSP. The plated wire memory and the external macro instruction memory are retained. The estimated characteristics are shown in Table 27. The figures in Table 27 do not include the plated wire memory, but typical numbers are included for the other memory types. The integrated circuit count and power analysis are shown in Table 28.

TABLE 27. BIT SLICE PROCESSOR CHARACTERISTICS

| | |
|---|---|
| Word Length | 24 bits |
| Technology | TTL LSI/MSI |
| Board Area | 115 $in.^2$ |
| Power | 28 W |
| IC's | 65 |
| Instructions | 64 |

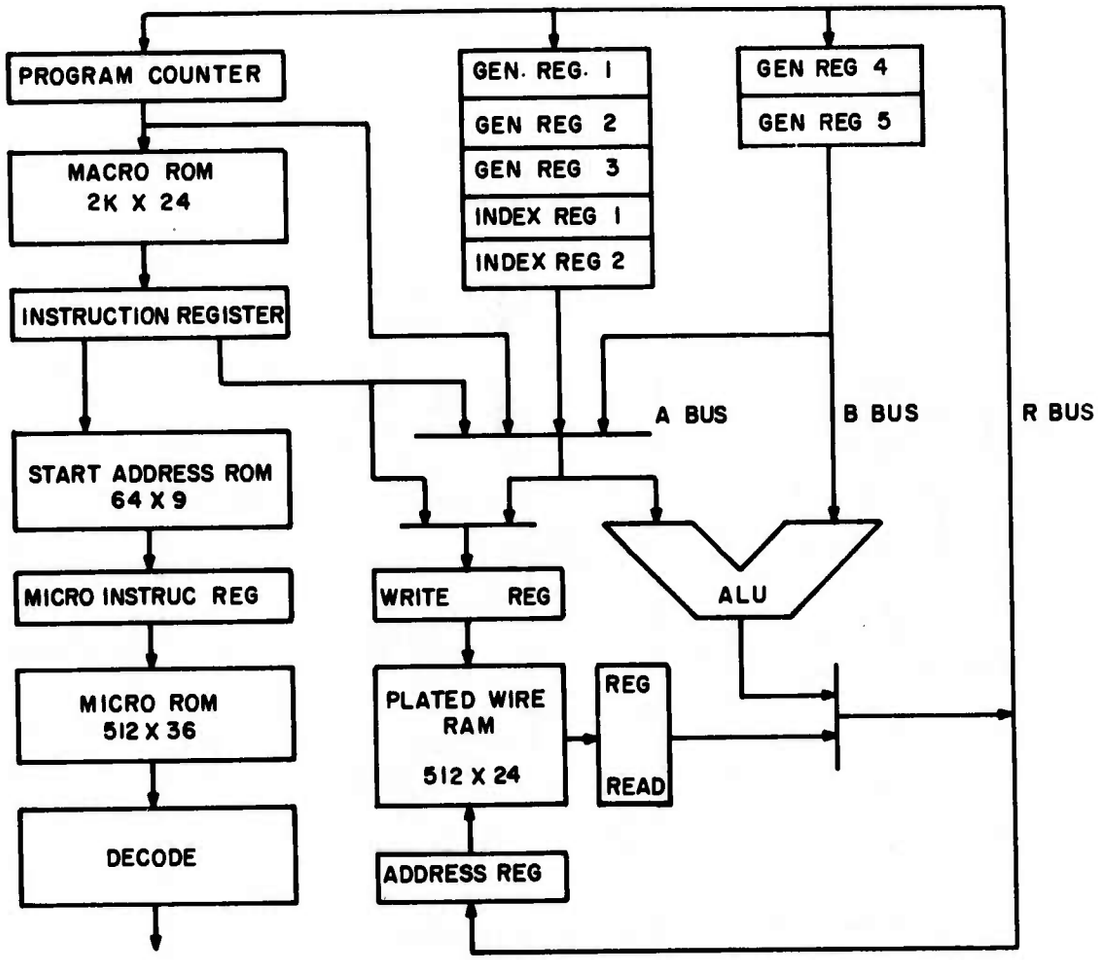



Figure 38. Navigation Computer Block Diagram

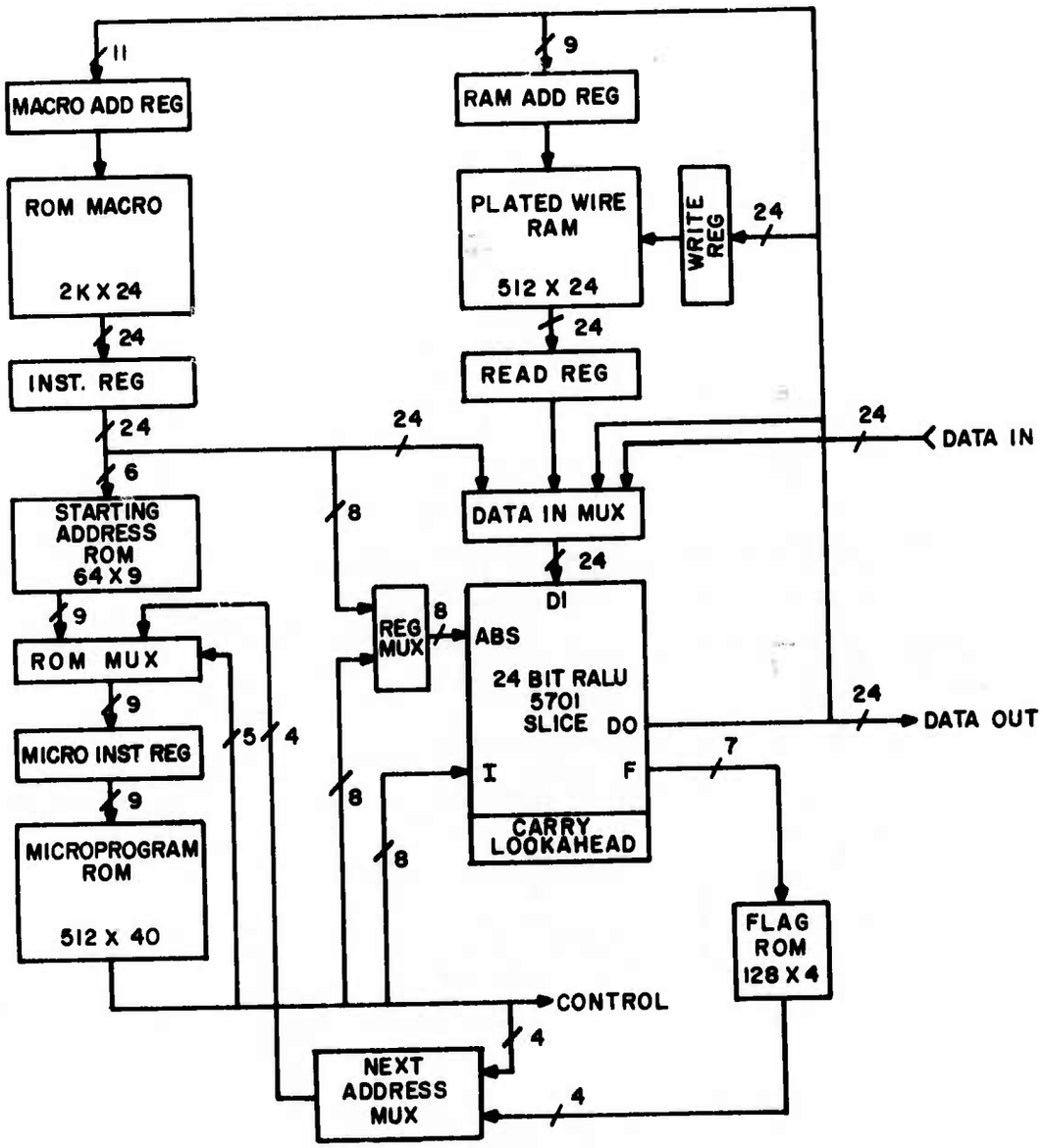



Figure 39. Bit Slice Processor Block Diagram

TABLE 28. BIT SLICE PROCESSOR CIRCUIT TYPES

| Function | Type | Quantity | Pins | Typical Power Each (mW) | Total (mW) |
|---|---|---|---|---|---|
| REGISTERS | | | | | |
| Macro Address, 11 bits | 54175 | 2 | 16 | 225 | 450 |
| Instruction, 24 bits | 54175 | 4 | 16 | 225 | 900 |
| PW Address, 9 bits | 54175 | 2 | 16 | 225 | 450 |
| PW Data Out, 24 bits | 54175 | 4 | 16 | 225 | 900 |
| PW Data In, 24 bits | 54175 | 4 | 16 | 225 | 900 |
| *Micro Address, 9 bits | 54175 | 2 | 16 | 225 | 450 |
| | Subtotal | 18 | | | 4050 |
| MULTIPLEXERS | | | | | |
| Data In, 4:1 | 54153 | 12 | 16 | 180 | 2160 |
| RALU Register, 2:1 | 54157 | 2 | 16 | 150 | 300 |
| *Next Address, 2:1 | 54157 | 1 | 16 | 150 | 150 |
| *ROM, 2:1 | 54157 | 3 | 16 | 150 | 450 |
| | Subtotal | 18 | | | 3060 |
| READ ONLY MEMORIES | | | | | |
| Macro, 2K x 24 | 3304 | 12 | 24 | 700 | 8400 |
| *Starting Address, 64 x 9 | 54186 | 2 | 24 | 300 | 600 |
| Micro, 512 x 40 | 3304 | 5 | 24 | 700 | 3500 |
| *Flag, 128 x 4 | 54187 | 1 | 16 | 470 | 470 |
| | Subtotal | 20 | | | 12970 |
| ARITHMETIC UNIT | | | | | |
| RALU, 24 bits | 5701 | 6 | 40 | 1150 | 6900 |
| Look-Ahead | 54182 | 3 | 16 | 180 | 540 |
| | Subtotal | 9 | | | 7440 |
| | Total IC's | 65 | | | |
| | Total Power | 27.5 W | | | |

*MCU parts

Both processors require a large investment in microprogram hardware. The six-bit operation code from the macro instruction memory is used to point to a starting address in the micro code. Next address information in the micro code can be influenced by arithmetic results such as overflow, all zeros, carry out, etc. A Microprogram Control Unit (MCU) containing the functions shown in Figure 40 would reduce the IC count and the power dissipation. The MCU contains the next address logic for the microprogram ROM. It examines the flags from the processor array and combines that information with the next address fields of the microcontrol word. If an MCU were available as a 40-pin DIP and dissipated 1.0 W, the IC count for the BSP would be reduced to 56, and the power dissipation to 26.7 W.

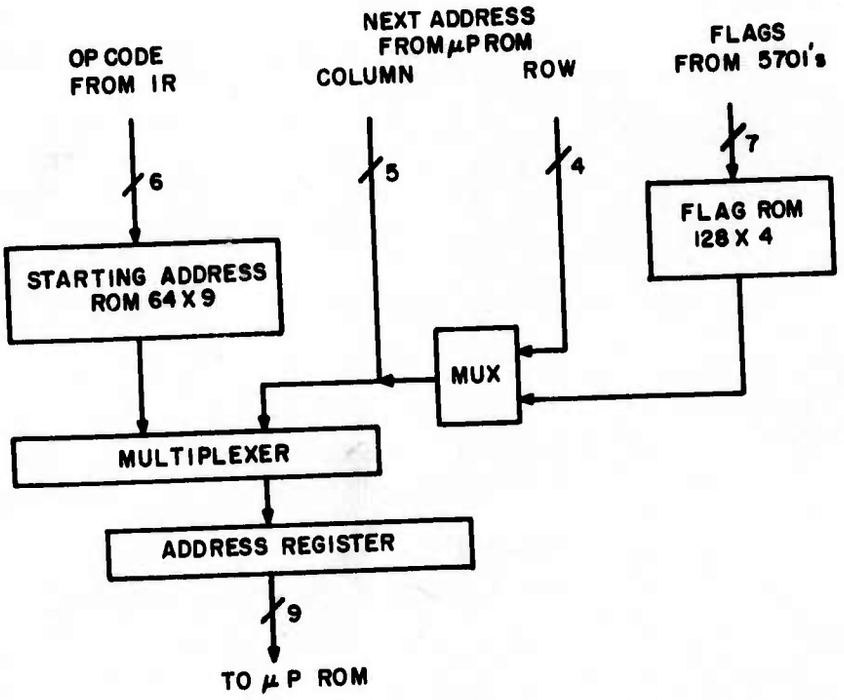



Figure 40. Microprogram Control Unit

3. General Purpose Processor

The BSP shown in Figure 39 does not have extensive input/output, direct memory access or interrupt facilities, which could be added as shown in Figure 41. These features would require additional hardware, board space, and power. An interrupt processor could provide the interface between device requests and the MCU. The MCU would provide mask and enable inputs to the interrupt processor, which, in turn, would provide a vector jump address to the MCU. The interrupt routine for the device would start at the jump address in the microprogram. The interrupt processor could contain priority encoders to allow the processor to respond quickly to the most critical input or output device. The I/O processor would provide the digital and analog interface to the peripheral devices. The PIA, such as those in the Motorola 6800 family, could be configured to serve the digital input and output. If the I/O processor cannot provide the required speed, the DMA processor can be used. The configuration shown in Figure 41 allows the peripheral to communicate with the data scratch pad, not the program store memory. The memory access controller arbitrates access requests between the DMA processor and the RALU processor.

The microprogrammable bit slice devices have placed LSI slices of registers and arithmetic logic units in the hands of the designer. As the users make the current suppliers aware of their needs for I/O, DMA, and interrupt processors, LSI devices suitable for these applications will also appear. In fact, some devices (PIA's, Modems, Universal Receiver/Transmitters) are presently available in the I/O area. Although the RALU slices presently become lost in the supporting logic, the situation should change in the future.

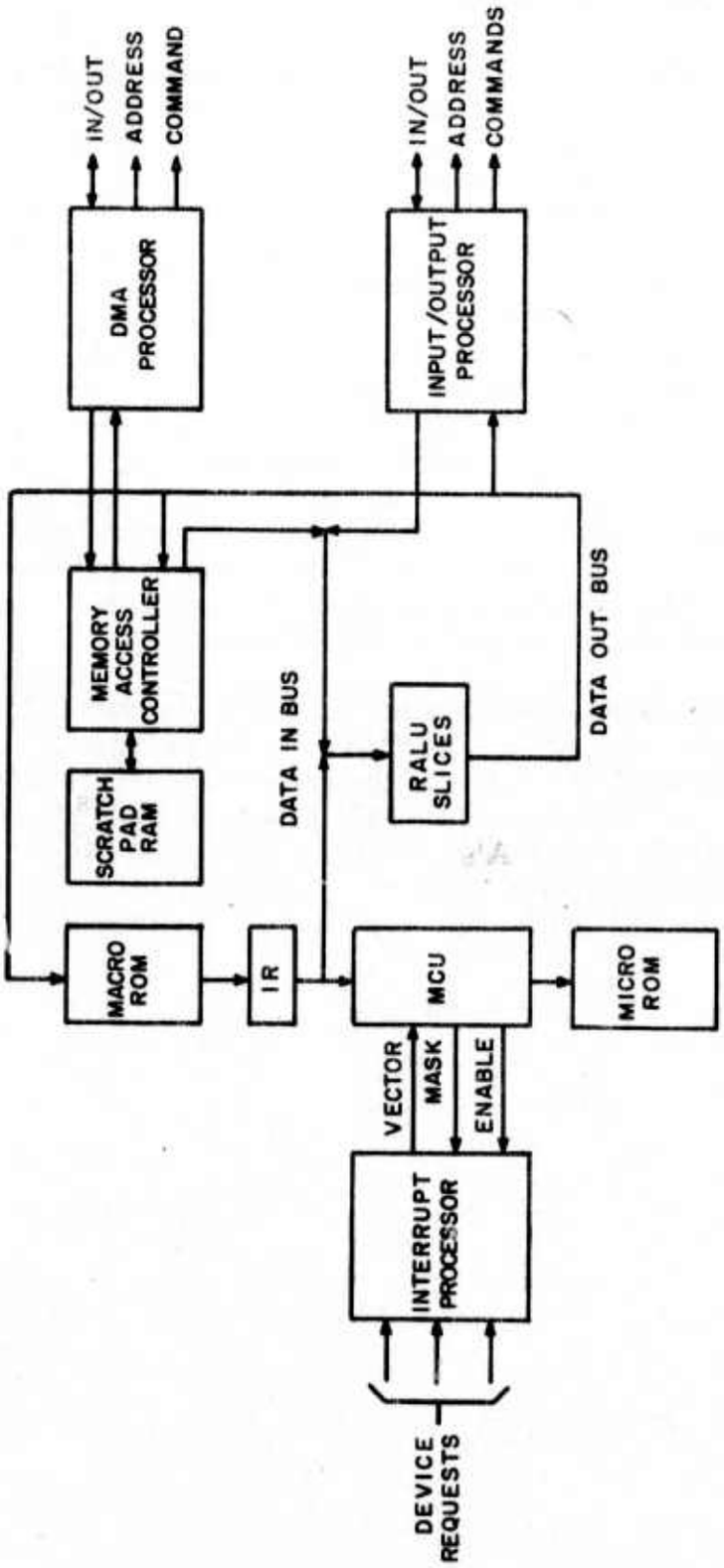



Figure 41. General Purpose Processor

## SECTION V

## SUMMARY AND CONCLUSIONS

The three application areas which were studied are only a small sample of potential areas for dedicated processors. Fixed word controllers were not examined. Preprocessors dedicated to one or more sensors controlling the analog to digital conversion, engineering unit conversion, and data formatting would not require sophisticated machines, but would relieve the downstream processor of tedious work.

The applications which were examined illustrate the wide range of processor requirements which could be served. Based on present technology, the bit slice microprocessor is the best vehicle for a standard part solution to the long word length application requirements. The present bit slice machines are microprogrammable, allowing the user to generate his own instruction set to tailor the processor to his own application needs. Alternatively, a standard instruction set could be provided with the ability to add special instructions at the option of the user.

The selection of a technology for a radiation hardened microprocessor depends on the radiation environment and the performance goals and their priorities. Dielectrically isolated TTL is the only current technology which will meet the most severe boost and re-entry radiation environments. Several technologies can meet the less severe airborne requirements, so that the selection becomes one based on performance criteria (speed, power), cost and availability, which are subject to tradeoffs by the user. The technologies of the currently available unhardened bit slice microprocessors are Schottky TTL and IIL.

There are two basic approaches to hardening electronic system designs. The first is to utilize the intrinsic hardness of the various existing technologies and specify that technology or mixture of technologies which will meet the requirements. The second is to use circuit design and processing techniques in the design and fabrication of custom, hardened components. The user may have no choice in accepting the cost, performance and risk penalties of the latter approach if his requirement cannot be met by existing technology.

Hardening requirements usually require speed-power tradeoffs in the device design. Device isolation will require additional silicon area so that there are fewer devices per unit area. As a further consequence, the total number of circuit nodes will increase, causing additional power dissipation and slowing down circuit speed. With lower device density, the hardened computer on a chip will be available at a later point in time than the unhardened

device. Until that point, the hardened computer will have to support an architecture which consists of a collection of simpler, functional components.

The recommended architecture for a radiation hardened microprocessor is based on a bit slice device. A four-bit slice is preferable over a two-bit slice for interconnect and area considerations. TTL technology provides the best combination of speed and power characteristics with the potential of the highest radiation hardness. The internal architecture of the slice should include multiple general registers, an accumulator extension register for multiply and divide operations, ALU flags such as carry, overflow, and zero for easy branch condition testing, and separate output ports for address and data. Microprogrammability would offer the user extensive flexibility, either in specifying a complete instruction set or in adding special instructions.

Several external LSI chips are desirable to support the bit slice chip and reduce the SSI/MSI device count. A Microprogram Control Unit (MCU) would interface between the macro instruction and microinstruction memories by examining the processor flags and other information to determine the address of the next microinstruction. An Interrupt Processing Unit (IPU) would examine the incoming service requests and generate address information to the MCU depending on processor state, priority and mask information. Input/Output Processors (IOP) would serve as digital channels between the I/O devices and the processor which would provide the IOP with commands and starting data and then return to other calculations. A Memory Access Controller would oversee the limited memory resource and grant requests to the DMA and IOP channels, and the processor.

These chips partition the microcomputer into functional LSI chips, which could then be configured in various quantities to satisfy specific application needs. A 16-bit microcomputer with the power of a present day minicomputer could be composed of eight LSI chips (four slices, an IOP, MAC, MCU and IPU), not including memory.

The attractiveness of microcomputers in terms of size and the resulting flexibility will eventually result in their use in space and airborne applications. A family of hardened microprocessors is desirable and technologically feasible, although their availability will be limited, and their costs high. A program to develop a hardened microprocessor will be least expensive if the design is based on an existing commercial product which has fully developed, documented support software.

## APPENDIX A

## REFERENCES


1. S. F. Schmidt et al, "Application of Kalman Filtering to the C-5 Guidance and Control System," Theory and Applications of Kalman Filtering, Chapter 13. NATO Publication, Feb. 1970 (AD-704-306).

2. J. M. Mendel, "Computational Requirements for a Discrete Kalman Filter," IEEE Trans. on Automatic Control, Dec. 1971.

3. L. Brock and G. T. Schmidt, "General Questions on Kalman Filtering in Navigation Systems," Theory and Applications of Kalman Filtering, Chapter 10. NATO Publication, Feb. 1970 (AD-704-306).

4. J. P. Raymond et al, Private Communication Data as Communicated by L. Creaser Motorola Inc., Semiconductor Products Division.

5. TAC-TIPS, Motorola Semiconductor Products Division, 30 April 1975.

6. TAC-TIPS, Motorola Semiconductor Products Division, 22 January 1975.

7. TAC-TIPS, Motorola Semiconductor Products Division, 30 May 1975.

8. J. E. Ashe, C. W. Perkins, T. J. Wang and F. T. Fujii, Study of Radiation Effects in MSI/LSI Technologies, Northrop Research Technology Center, Report NRTC 74-6R, January 1974.

9. R. M. Orndorff, E. L. Hammer, and N. A. Graf, Radiation Testing of Hardened Electronic Components, Hughes Aircraft Company, Final Report, Contract F04701-71-C-0115. SAMSO Report TR71-301, December 1971.

10. R. W. Marshall and D. K. Meyers, Studies of the Nuclear Radiation Performance of Isoplanar Integrated Circuits and Circuit Components. Fairchild Camera and Instrument Corp., Contract Number DAAG 39-73-C-0002, for Harry Diamond Laboratories, July 1974.

11. J. P. Raymond, D. N. Pocock, C. W. Perkins, and J. E. Ashe, MSI/LSI Vulnerability Study, Northrop, Final Report for DNA Contract No. DASA-01-70-C-0093, Northrop Report NRTC 73-11R, March 1973.

12. D. E. Berglund and W. J. Patterson, Design of a Hybrid MNOS/Plated Wire Memory System, General Electric Company, AESD, Technical Report on Contract No. F33615-74-C-1096, for AFAL, Report AFAL-TR-75-60, 1 April 1975.

13. S. Tietelbaum (RCA), Private Communication at Hardened Guidance and Weapon Delivery Technology Meeting, held at Boeing, Seattle, Washington, 22-24 April 1975.

14. Private Communication with R. B. Buchanan, AFCRL, Hanscombe Field, Mass.



☆ U. S. GOVERNMENT PRINTING OFFICE: 1976 — 657-650/576